

OBSAH

Jiří Veselý: Dekadická soustava – úvodní poznámka	1
Sebastian Rahtz, Michel Goossens: Příručka $\text{T}_{\text{E}}\text{X}$ Live, piate vydanie . . .	3
Jiří Tesař: DVIBRAILLE	44
David Antoš: $\text{T}_{\text{E}}\text{X}$ Versioning System aneb jak všechny zdrojové soubory uložit	47
Zdeněk Wagner: Spolupráce databáze s $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ em	49
Tomáš Hála: Jedna z možností databázového publikování v $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ u	78
Josef Svoboda: Dojmy z 19. bienále grafického designu v Brně	84
Zuzana Došlá, Roman Plch a Petr Sojka: CD-ROM Matematická analýza s programem Maple: 1. Diferenciální počet funkcí více proměnných .	86
Karel Skoupý: Zpráva o prezentaci $\mathcal{N}\mathcal{T}\mathcal{S}$ na TUG'2000.	87

Toto číslo obsahuje CD-ROM $\text{T}_{\text{E}}\text{X}$ Live 5.

Zpravodaj Československého sdružení uživatelů $\text{T}_{\text{E}}\text{X}$ u je vydáván v tištěné podobě a distribuován zdarma členům sdružení. Po uplynutí dvanácti měsíců od tištěného vydání je poskytován v elektronické podobě (PDF) ve veřejně přístupném archívu dostupném přes <http://www.cstug.cz>.

Své příspěvky do Zpravodaje můžete zasílat v elektronické podobě anonymním ftp na <ftp.icpf.cas.cz> do adresáře `/wagner/incoming/`, nejlépe jako jeden archivní soubor (`.zip`, `.arj`, `.tar.gz`). Současně zašlete elektronickou poštou upozornění na <mailto:bulletin@cstug.cz>. Uvedený adresář je přístupný pouze pro zápis. Pokud nemáte přístup na Internet, můžete zaslat příspěvek na disketě na adresu:

Zdeněk Wagner
Vinohradská 114
130 00 Praha 3

Disketu formátujte nejlépe pro DOS, formáty Macintosh 1.44 MB a EXT2 jsou též přijatelné. Nezapomeňte přiložit všechny soubory, které dokument načítá (s výjimkou standardních součástí $\mathcal{C}_{\text{S}}\text{T}_{\text{E}}\text{X}$ u), zejména v případě, kdy vás nelze kontaktovat e-mailem.

ISSN 1211-6661

Rád bych připomněl jedno výročí: letos v květnu uplynulo již deset let od vzniku Československého sdružení uživatelů $\text{T}_{\text{E}}\text{X}$ u (ζTUG). Každé narozeniny svádí k bilancování, ale u malých jubileí nebývá obvyklé; doufám, že ζTUG tak hned nezanikne a že ještě bude dostatek příležitostí k bilancování povolanějšími a později.

Ještě před rokem 1989 bylo jasné, že sazba matematických textů na západní polokouli prošla revoluční změnou. Když se nám po náročných jednáních podařilo získat pro Matematický ústav UK na MFF za cca čtvrt milionu dvě PC s jehličkovou tiskárnou, mohli jsme pomýšlet i na změnu sazby našeho časopisu. Není příliš překvapující, že jsme dospěli jednoznačně k jediné možnosti a že jí byl $\text{T}_{\text{E}}\text{X}$.

Po bližším seznámení s $\text{T}_{\text{E}}\text{X}$ em se dostavila zákonitě euforie a fascinace jeho schopnostmi. Cítili jsme touhu i povinnost ho propagovat a šířit. Tak vznikl ζTUG jako potenciální platforma pro matematiky, techniky, filology, přírodovědce, programátory, překladatele, typografy, redaktory časopisů, pracovníky nakladatelství, zaměřených na technicky náročné texty: $\text{T}_{\text{E}}\text{X}$ se totiž jeví jako křižovatka, na níž se jejich cesty sbíhají. Měla to být platforma pro přátelské diskuse, založené na vzájemném pochopení a soustředěném zájmu o moderní techniku sazby složitých a vzhledově krásných textů. Připomeňme na tomto místě stručně historii $\text{T}_{\text{E}}\text{X}$ u.

Je zde patrná jistá spirála vývoje: vývoj matematiky pomohl vytvářet tlak na změnu technologie tvorby knih, různorodost matematických značek a odlišnost matematické sazby se však v určité etapě stala pro typografii obtížnou přítěží. Byla to však v širším smyslu opět matematika, která přinesla řešení situace. $\text{T}_{\text{E}}\text{X}$ vznikl v USA v letech 1978–1982 a v krátké době se rozšířil po celém světě, zejména v akademických kruzích. Zrod jeho popularity souvisí s jeho geniální koncepcí, jejímž autorem je Donald Knuth, i s postojem Americké matematické společnosti k $\text{T}_{\text{E}}\text{X}$ u; ta na $\text{T}_{\text{E}}\text{X}$ navázala rozsáhlý program elektronizace matematických časopisů. Po dokončení ho Knuth daroval veřejnosti: $\text{T}_{\text{E}}\text{X}$ je volně šiřitelný program a je proto relativně snadno dostupný.

Zmíňme ještě další výročí: Dne 22. února 1980 byla založena organizace uživatelů $\text{T}_{\text{E}}\text{X}$ u, (TUG — $\text{T}_{\text{E}}\text{X}$ Users Group); téhož roku začal vycházet i TUG -boat, časopis věnovaný plně problematice $\text{T}_{\text{E}}\text{X}$ u. Rád bych připomněl na přelomu století význam vytvoření $\text{T}_{\text{E}}\text{X}$ u citováním výroku Gordona Bella: *Don Knuth's [$\text{T}_{\text{E}}\text{X}$] is potentially the most significant invention in typesetting in this century. (...) in terms of importance could rank near the introduction of the Gutenberg*

print. [T_EX Dona Knutha je potenciálně nejdůležitějším objevem tohoto století v oblasti typografie (...) a co do důležitosti ho lze srovnat s Gutenbergovým knihtiskem.]

Věřím, že ζ TUG své základní poslání stále plní. Má za sebou řadu úspěchů, ale i některé prohry. Připomeňme na jedné straně existenci české a slovenské implementace pro mnoho platform, existenci (bohužel již zastaralého) programu pro korekci pravopisu, propojení na ostatní sdružení uživatelů T_EXu ve světě, existenci řady časopisů, které by bez T_EXu patrně zanikly i vlastního Zpravodaje, relativně dobrou současnou finanční situaci Sdružení. Zpravodaj má nejen velmi dobrou grafickou úroveň, je však i zajímavý a žádoucí nejen pro naše T_EXové společenství: dva články vyšly dokonce ve Zpravodaji dříve nežli se objevil v TUGboatu jejich vyžádaný překlad. Máme i českou literaturu o T_EXu, umožňující za cenu jisté námahy připojit se k naší komunitě.

Na druhou miskou vah je však třeba položit to, že se potýkáme s dědictvím „československého fenoménu“: nejsou lidi. Sdružení je obětavě řízeno dobrovolníky, kteří často zoufale zápasí s časem. Všichni užíváme dobrodiní T_EXu v mnoha implementacích, které však nejsou vždy zcela kompatibilní. Nemáme standardní implementace pro začátečníky, potřebovali bychom nový korektor pravopisu. I samotná zdánlivě rutinní práce spojená s údržbou toho, co máme, vyžaduje nezanedbatelnou práci, za kterou sdružení vděčí obětavým jedincům.

I přes naše společné úsilí u nás stále platí, že v problematice následného zpracování textů v elektronické formě v komerčních tiskárnách podstatně zaostáváme za současným světovým standardem. Nezbyvají nám také síly na vyvíjení různých maker, podpůrných programů, vydávání dalších potřebných textů.

Zmínil jsem pár věcí: za sebe si v tuto chvíli slibuji, že své, snad ne velike (pracovní, nikoli finanční) dluhy vůči ζ TUGu brzo vyrovnám. A ζ TUGu při této příležitosti přeju, aby to udělali i ostatní, kterým je T_EX blízký. A to by mohli být patrně všichni ti, kteří tento úvodník čtou.

Jiří Veselý

Príručka T_EX Live, piate vydanie

SEBASTIAN RAHTZ, MICHEL GOOSSENS

Obsah

1	Úvod	4
1.1	Rozšírenia T _E Xu	4
2	Štruktúra a obsah CD-ROM	5
2.1	TDS strom	5
3	Inštalácia a použitie pod operačným systémom Unix	6
3.1	Spúšťanie T _E X Live z CD-ROM	6
3.2	Inštalácia T _E X Live na pevný disk	8
3.3	Inštalovanie jednotlivých balíkov T _E X Live na pevný disk .	11
3.4	Program texconfig	12
4	Inštalácia a použitie pod operačným systémom Windows	12
4.1	Spúšťanie T _E X Live z CD-ROMu	12
4.2	Inštalácia na pevný disk	13
5	Budovanie na novej platforme Unixu	13
5.1	Predpoklady	14
5.2	Konfigurácia	14
5.3	Spúšťanie make	14
5.4	Záverečné kroky konfigurácie	15
6	Používateľská príručka ku systému Web2c	15
6.1	Vyhľadávanie ciest knižnicou Kpathsea	17
6.2	Súborové databázy	21
6.3	Možnosti nastavenia za behu programu	30
7	História a poďakovania	31
8	Budúce verzie	33
9	Súbor texmf.cnf	33
	Summary: The T_EX Live Manual, 5th Edition	43

1. Úvod

Táto dokumentácia popisuje hlavné črty **T_EX Live 5** CD-ROM—T_EX/L^AT_EX distribúciu pre Unix a Windows32 systémy, ktorá zahŕňa T_EX, L^AT_EX 2_ε, META-FONT, METAPOST, Makeindex and BIB_TE_X; širokú množinu makriér, fontov a dokumentácie prispôsobenú ‘*T_EX Directory Standard*’ (TDS), ktorá môže byť použitá takmer s každým nastavením T_EXu.

Tento T_EX balík používa modifikovanú Web2c (verzia 7.3) implementáciu programov, ktorá sa snaží urobiť prácu s T_EXom tak ľahkú, ako je len možné a využíva všetky výhody efektívnej a vysoko prispôsobiteľnej knižnice Kpathsea od Karla Berryho and Olafa Webera. T_EX môže byť spúšťaný buď priamo z CD-ROMu alebo byť nainštalovaný na pevný disk.

Väčšinu zo spustiteľných systémov na CD-ROMe tvoria ovládače a podporné programy pre T_EX, vrátane *dvips* (prekladača z DVI do Postscriptu), *xdvi* (X Windows prehliadač), *dvilj* (HP LaserJet ovládač), *lacheck* (kontrola syntaxe L^AT_EXu), *tex4ht* konvertor (preklad z T_EXu do HTML formátu), *dviconcat* a *dviselect*, *dv2dt* a *dt2dv* (*dvi* konvertor do ASCII a opačne) a postscriptové nástroje od Angusa Duggana.

1.1. Rozšírenia T_EXu

Spúšťateľné systémy **T_EX Live** obsahujú tri experimentálne rozšírenia štandardného T_EXu:

1. ε -T_EX, ktorý pridáva malú, ale výkonnú množinu nových primitívov a T_EX–X_EL rozšírenia pre sadzbu zľava doprava; v základnom móde, ε -T_EX je 100% kompatibilný s bežným T_EXom. Bližšie detaily nájdete na CD-ROMe v súbore `texmf/doc/etex/base/etex_man.pdf`.
2. pdfT_EX, ktorý ponúka možnosť zapisovať vo formáte Acrobat PDF namiesto DVI. Príručku používateľa nájdete v `texmf/doc/pdftex/pdftex-1.pdf`. V súbore `texmf/doc/pdftex/samplepdf/samplepdf.tex` nájdete príklad použitia. L^AT_EX *hyperref* balík ponúka voľbu `,pdftex`, ktorá zapne všetky funkcie programu.
3. Ω (Omega), ktorá pracuje vnútorne so 16-bitovými znakmi, používajúc Unicode; toto jej umožňuje pracovať s takmer všetkými svetovými skriptami simultánne. Takisto podporuje dynamicky nahrávané ‘ Ω Translation Processes’ (OTPs), ktoré umožňujú používateľom definovať komplexné transformácie, ktoré majú byť vykonané na ľubovoľných tokoch vstupu.

Dokumentáciu nájdete v súbore `texmf/doc/omega/base/doc-1.8.tex`.

Verzia ε -T_EXu (2.1) je konečná, hoci v budúcich vydaniach pribudnú nové funkcie. pdfT_EX (verzia 0.14f) a Ω (verzia 1.11) sú v neustálom vývoji, verzie na tomto CD-ROMe sú aktuálne verzie z polovice marca 2000.

2. Štruktúra a obsah CD-ROMu

Nasledujúci zoznam obsahuje najdôležitejšie koreňové adresáre na CD-ROMe:

bin Hlavné $\text{T}_{\text{E}}\text{X}$ programy, zoskupené do adresárov podľa platforiem.

tldoc Dokumentácia pre $\text{T}_{\text{E}}\text{X Live}$.

FAQ Často kladené otázky (Frequently Asked Questions) v angličtine, francúzštine a nemčine.

info Dokumentácia v GNU ‚info‘ formáte o systéme $\text{T}_{\text{E}}\text{X}$.

man Dokumentácia vo forme Unix ‚man‘ stránok o systéme $\text{T}_{\text{E}}\text{X}$.

source Zdrojové súbory všetkých programov, vrátane hlavných distribúcií Web2c, $\text{T}_{\text{E}}\text{X}$ a METAFONT distribúcie. Tieto sú uložené v skomprimovanom tar archíve.

support Rôzne časti $\text{T}_{\text{E}}\text{X}$ -príbuzného softwaru, ktoré *nie* sú inštalované automaticky, ako napríklad Musix $\text{T}_{\text{E}}\text{X}$, podporné programy a úplná distribúcia Ghostscriptu, verzia 5.50.

texmf Hlavný podporovaný strom s makrami, fontami a dokumentáciou.

usergrps Materiál o $\text{T}_{\text{E}}\text{X}$ User Groups.

Nachádzajú sa tu takisto dva inštalačné skripty pod systém Unix:

`install-cd.sh` a `install-pkg.sh`; budeme sa nimi zaoberať v odseku 3 na str. 6.

2.1. TDS strom

$\text{T}_{\text{E}}\text{X Live}$ `texmf` strom pozostáva z rôznych kolekcií, z ktorých každá obsahuje množstvo balíkov, ktorých sa nachádza na CD-ROMe vyše 400. Normálna inštalácia umožňuje používateľovi prekopírovať všetky kolekcie na lokálny pevný disk z CD-ROMu, ale takisto je možné inštalovať len jeden balík z kolekcie. Tu je zoznam kolekcií:

ams Balíky makier a fontov od American Mathematical Society.

bibtex BIB $\text{T}_{\text{E}}\text{X}$ štýly a databázy.

doc Všeobecné príručky a dokumentácia v rôznych formátoch, vrátane HTML a PDF.

dvips Podpora pre Rokického ‚DVI -to-PostScript‘ ovládač.

etex Podpora ε - $\text{T}_{\text{E}}\text{X}$ u.

fonts Zdrojové súbory fontov, metrik, PostScriptových a bitmapových formátov.

formats Eplain, Rev $\text{T}_{\text{E}}\text{X}$, phyzzx, texsis, alatex, text1, lollipop, atď.

generic Makrá navyše použiteľné s ľubovoľným formátom.

graphics Balíky makier pre grafiku.

lang Jazyková podpora pre iné ako anglické jazyky.

latex L $\text{T}_{\text{E}}\text{X}$, zahŕňajúci oficiálne nástroje a všetky balíky podporujúce L $\text{T}_{\text{E}}\text{X} 2_{\varepsilon}$.

metapost Podpora pre METAPOST.
omega Podpora pre projekt Ω .
pdftex Podpora pre pdf \TeX .
plain Makrá pre plain \TeX .
systems Binárne súbory pre Unix a Win32 platformy.
texlive Základný materiál pre distribúciu.

Každá z kolekcií je rozdelená do *základnej* (1), *odporúčanej* (2) a *zvyšnej* (3) časti. Teda všetky balíky v kolekcii **latex1** sú tie, ktoré sú nutné na prácu s \LaTeX om, balíky v kolekcii **latex2** sú odporúčané väčšine používateľov a kolekcia **latex3** obsahuje dodatočne voliteľné balíky. Adresár **texmf/tpm** obsahuje zoznamy všetkých súborov v každom balíku (používané inštaláčnymi programami).

3. Inštalácia a použitie pod operačným systémom Unix

\TeX Live CD-ROM možno použiť tromi spôsobmi:

1. Môžete pripojiť CD-ROM na váš súborový systém, prispôbiť vašu **PATH**, a spúšťať všetko z CD-ROMu; toto riešenie vyžaduje veľmi málo diskového priestoru a umožňuje vám okamžitý prístup ku všetkým dátam na CD-ROMe. Hoci výkonnosť nebude optimálna, je toto riešenie vhodné, napríklad v prípade PC počítačov, na ktorých beží operačný systém Linux.
2. Je možné nainštalovať celý systém alebo jeho časť na váš lokálny pevný disk. Toto je najlepšia metóda pre väčšinu ľudí, ak majú dostatok miesta na disku (minimum okolo 10 megabajtov, okolo 100 megabajtov na odporúčaný systém).
3. Je možné nainštalovať vybrané balíky na prácu buď s vašim už existujúcim \TeX systémom alebo **\TeX Live** systémom, ktorý ste nainštalovali predtým.

Každá z horeuvedených metód je bližšie popísaná v nasledujúcich odsekoch.

3.1. Spúšťanie **\TeX Live** z CD-ROMu

Usporiadanie Web2c vám dovoľuje spúšťať programy jednoducho pridaním zodpovedajúceho adresára v adresári **bin** na CD-ROM do vašej **PATH**. Podporné súbory budú potom nájdené bez toho, aby ste robili čokoľvek ďalšie. Nasledujúca tabuľka zobrazuje zoznam možných systémov a im zodpovedajúce adresáre.

Upozornenie: Tento CD-ROM je v ISO 9660 (High Sierra) formáte s rozšíreniami Rock Ridge a Joliet. Aby ste využili všetky výhody tohoto CD-ROMu v systéme Unix, váš systém musí byť schopný využívať rozšírenia Rock Ridge. Prečítajte si prosím dokumentáciu k príkazu `mount` a presvedčte sa, či je to možné. Ak máte viacero rôznych počítačov pripojených do lokálnej siete, presvedčte sa, či môžete umiestniť CD-ROM na ten z nich, ktorý *podporuje* Rock Ridge a použiť ho s ostatnými.

Linux, FreeBSD, Sun, SGI a DEC Alpha systémy by mali byť schopné používať CD-ROM bez problémov. Kvôli budúcim verziám tejto dokumentácie by sme ocenili detailné pripomienky užívateľov ostatných systémov.

Nižšie uvedený postup inštalácie predpokladá, že ste už úspešne nainštalovali CD-ROM s úplnou Rock Ridge kompatibilitou.

DEC Alpha ev5 OSF 4.0d	alphaev5-osf4.0d
HP9000 HPUNIX 10.10	hppa20-hpux10.20
Intel x86 s GNU/Linux	i386-linux
Intel x86 s FreeBSD ELF 3.4	i386-freebsd
SGI IRIX 6.5	mips-irix6.5
IBM RS 6000 AIX 4.2.*	rs6000-aix4.2.1.0
Sun Sparc Solaris 2.7	sparc-solaris2.7
Windows 9X/2000/NT	win32

Možno vás trápí otázka, čo sa stane s fontami alebo zmenenou konfiguráciou, ktorá vznikne pri vašej práci, keďže nemôžete meniť súbory na CD-ROMe. Avšak, vy si môžete udržiavať paralelný, zapisovateľný \TeX strom na vašom pevnom disku. Tento bude prehľadávaný skôr ako hlavný strom na CD-ROMe. Predvolené umiestnenie je `texmf-localconfig` na CD (ktoré neexistuje!), čiže toto nastavenie *musíte* zmeniť nastavením premennej `VARTEXMF`.

Používatelia `sh` alebo `bash` na Intel PC so systémom Linux môžu pripojiť \TeX Live CD-ROM na `/cdrom` napísaním príkazu:

```
>> mount -t iso9660 /dev/cdrom /cdrom
```

Ďalej je treba pridať do premennej `PATH` adresár obsahujúci binárne súbory pre danú architektúru medzi prehľadávané adresáre.

```
PATH=/cdrom/bin/i386-linux:$PATH
```

```
export PATH
```

```
VARTEXMF=/usr/TeX.local
```

```
export VARTEXMF
```

Je vhodné tieto riadky vložiť aj do skriptu `.profile`.

Ak máte pochybnosti, obráťte sa na vášho lokálneho systémového administrátora so žiadosťou o pomoc pri inštalácii vášho CD-ROMu alebo konfigurácii adresárov.

Požadované podporné súbory budú inštalované na váš pevný disk vtedy, keď ich budete prvý raz potrebovať. Dobrým nápadom je spustiť skript `texconfig` okamžite a presvedčiť sa, že všetko funguje.

3.2. Inštalácia T_EX Live na pevný disk

Všetky potrebné kroky na inštaláciu celej alebo časti distribúcie na váš pevný disk zahŕňajú pripojenie CD-ROMu, prejdienie do koreňového adresáru a napísanie príkazu:

```
>> sh install-cd.sh
```

(Na niektorých systémoch Unix možno budete musieť použiť `sh5` alebo `bash`.) Tento skript pristupuje k zoznamom kolekcií a balíkov na CD-ROMe a snaží sa zistiť, na akom počítačovom systéme pracujete. Mal by začať zobrazovať nasledujúce riadky:

```
Initializing collections... Done initializing.  
Counting selected collections... Done counting.  
Calculating disk space requirements for collections...Done calculating that.  
Initializing system packages... Done initializing system.
```

Potom zobrazí hlavnú kontrolnú obrazovku (Figure 1), ktorá vám umožňuje meniť štyri veci:

1. typ systému, ktorý máte, alebo pre ktorý chcete inštalovať;
2. kolekcie, ktoré chcete inštalovať a úroveň inštalácie (*základná*, *odporúčaná* alebo *vyššia*);
3. lokalizácia na vašom pevnom disku, kde majú byť súbory umiestnené;
4. niektoré vlastnosti za behu programov;

Možnosť si vyberáte napísaním písmena alebo čísla a stlačením `,return'`. V príklade bol detekovaný systém Linux ELF, bol nastavený predvolený typ úrovne inštalácie pre všetky kolekcie *odporúčanej* (recommended) a predvolený inštalačný adresár je `/usr/local`. Všimnite si, že je taktiež zobrazené miesto na pevnom disku potrebné na inštaláciu pri súčasnom nastavení. Ak by ste akceptovali navrhované nastavenie, budete potrebovať okolo 100 megabajtov voľného diskového priestoru. Avšak, základné nastavenie by zabralo iba okolo 10 megabajtov a možno ho rozšíriť vybranými balíkmi vtedy, keď ich budete potrebovať.

V adresári, ktorý ste zvolili na inštaláciu, umiestni inštalačný skript binárne súbory do podadresáru `bin` a podporný súborový strom do podadresáru `texmf`.

Ponuky `options` vám umožňujú sa rozhodnúť, či sa nové fonty budú vytvárať v inom adresári (ak chcete, aby bol hlavný balík pre väčšinu užívateľov prístupný iba na čítanie) a či budú vytvorené symbolické spojenia pre informačné stránky `man` a `GNU info` v štandardných adresároch; budete samozrejme potrebovať správcovské práva, aby ste to urobili.

```

=====> TeX Live installation procedure <=====
===> Note: Letters/digits in <angle brackets> indicate menu items <===
===>         for commands or configurable options                               <===

Proposed platform: Intel x86 with GNU/Linux
<P> over-ride system detection and choose platform
<C> collections:      24 out of 34, disk space required: 9812099 kB
<S> systems:         1 out of 8, disk space required: 7925 kB
                        total disk space required: 9820024 kB
<L> install level (1: basic, 2: recommended, 3: all): 2
<D> directories:
    TEXDIR      (The main TeX directory)           : /usr/TeX
    TEXMFLOCAL (TeX directory for local styles etc): /var/TeX-local
<O> options:
    [ ] alternate directory for generated fonts ()
    [ ] alternate directory for configuration ()
    [ ] create symlinks in standard directories
    [ ] do not install macro/font doc tree
    [ ] do not install macro/font source tree
    [ ] only install free software
<I> start installation, <H> help, <Q> quit
Enter command:

```

Obr. 1: Hlavná kontrolná obrazovka

```

name                selection      size
<1> bibtex          [recommended]  7597 kB
<2> doc             [recommended] 21152 kB
<3> dvips           [recommended]   430 kB
<4> etex            [recommended]   102 kB
<5> fonts           [recommended] 51447 kB
<6> formats         [recommended] 14651 kB
<7> generic         [recommended]   459 kB
<8> graphics        [recommended]  9674 kB
<9> lang            [recommended] 19618 kB
<U> latex           [recommended] 23429 kB
<V> metapost        [recommended]  1443 kB
<W> omega           [recommended]  4986 kB
<X> pdftex          [recommended]   471 kB
<Y> plain           [recommended]  1113 kB
<Z> texlive        [recommended] 10155 kB
                        SUM: 166829 kB
=====
global commands: select <N>one / <B>asic / R<E>commended / <A>ll
                  for all collections
<R>  return to platform menu
<Q>  quit

```

Obr. 2: Výber kolekcií

```

Collection: Fonts
=====

Fonts, including metrics, virtual fonts and sources

=====
<N> No packages
<B> Basic packages           [ 1023 kB]
<E> Basic + Recommended packages [ 51447 kB]
<A> All packages            [127417 kB]

=====
<R> return to collection menu
<Q> quit

Enter command:

```

Obr. 3: Prispôsobenie kolekcie

Keď zvolíte <C> ako voľbu pre kolekcie, zobrazí sa obrazovka s výberom možných kolekcíí, predvolená úroveň inštalácie a požadované voľné miesto na disku (obrázok 2). Môžete si zvoliť požadovanú úroveň inštalácie pre každú kolekciu, z možných volieb od *none* (žiadnej) až po *all* (kompletnú). Toto nastavenie môžete urobiť buď pre všetky kolekcie naraz, alebo si zvolíte určitú kolekciu a nastaviť jej úroveň (obrázok 3).

Keď ste hotoví, vráťte sa na hlavnú obrazovku a odštartujte inštaláciu. Inštaláčny program zoberie každú kolekciu a systémy, ktoré ste si zvolili, prezrie zoznam súborov na CD-ROMe a zostaví hlavný zoznam súborov, ktoré sa budú prenášať. Tieto potom budú prekopírované na váš pevný disk. Ak ste inštalovali systém, vyvolá sa jeho inicializačný program (vytváranie formátových súborov, atď.). Keď sa táto fáza skončí posledné, čo potrebujete spraviť, je pridať správny podadresár `bin` z vašej inštalácie `TEXu` do vašej `PATH` a môžete začať používať `TEX`. Ak chcete, môžete premiestniť binárne súbory o jednu úroveň vyššie, napr. z adresára `/usr/local/bin/alpha-osf3.2` do adresára `/usr/local/bin`. Ak to však spravíte, musíte editovať súbor `texmf/web2c/texmf.cnf` (pozri Prílohu 9) a upraviť riadok nikde na začiatku, ktorý číta

```
TEXMFMAIN = $SELFAUTOPARENT
```

na

```
TEXMFMAIN = $SELFAUTODIR
```

Ak premiestnite binárne súbory do úplne iného adresárového stromu, budete musieť editovať `TEXMFMAIN` a špecifikovať adresárový strom podporných súborov explicitne a nastaviť `TEXMFCNF` vo vašom prostredí na `$TEXMFMAIN/texmf/web2c`.

3.3. Inštalovanie jednotlivých balíkov T_EX Live na pevný disk

Niekedy možno budete potrebovať opäť použiť T_EX Live CD-ROM buď na aktualizáciu už existujúcej inštalácie alebo na pridanie nových programov do už existujúcej inštalácie z CD-ROMu. Keďže hlavný inštalačný program je určený iba na prvotnú inštaláciu, na dodatočné inštalácie je možné použiť skript `install-pkg.sh` z CD-ROMu. Spustíte ho pripojením CD-ROMu, prejdením do pripojeného adresáru a spustením príkazu

```
>> sh install-pkg.sh options
```

Skript je možné spustiť s deviatimi nastaveniami. Prvé štyri umožňujú určiť: meno individuálneho balíka, ktorý chcete nainštalovať, meno celej kolekcie (napríklad `ams2`), meno pripojeného adresáru CD-ROMu a meno adresáru obsahujúceho zoznamov súborov (bežne sú tieto posledné dve hodnoty nastavené automaticky):

```
--package=name  
--collection=name  
--cddir=name  
--listdir=name
```

Čo sa skutočne bude diať určujú ďalšie štyri nastavenia; prvé dve vám umožňujú vypustiť z inštalácie dokumentáciu a zdrojové súbory, tretie zastaví predvolené spúšťanie `mktexlsr` na záver inštalácie, ktoré prebuduje databázu súborov a štvrté nerobí nič len vypíše zoznam súborov, ktoré budú inštalované:

```
--nodoc  
--nosrc  
--nohash  
--listonly
```

Nakoniec, posledné nastavenie umožňuje určiť, že namiesto inštalácie súborov skript vytvorí iba tar archív v špecifikovanom adresári:

```
--archive=name
```

Preto, keby sme napr. jednoducho chceli vidieť súbory, ktoré tvoria balík `fancyhdr` skôr, ako ho nainštalujeme, náš príkaz a následný výstup by mohol vyzeráť takto:

```
>> sh install-pkg.sh --package=fancyhdr --listonly
```

```
texmf/doc/latex/fancyhdr/fancyhdr.dvi  
texmf/doc/latex/fancyhdr/fancyhdr.tex  
texmf/lists/latex3/fancyhdr  
texmf/source/latex/fancyhdr/README  
texmf/source/latex/fancyhdr/fancyheadings.new  
texmf/tex/latex/fancyhdr/extramarks.sty  
texmf/tex/latex/fancyhdr/fancyhdr.sty  
texmf/tex/latex/fancyhdr/fixmarks.sty
```

Iné príklady použitia:

- Nainštaluj L^AT_EXový balík `natbib`:
`>> sh install-pkg.sh --package=natbib`
- Nainštaluj L^AT_EXový balík `alg` bez zdrojových súborov a dokumentácie:
`>> sh install-pkg.sh --package=alg --nosrc --nodoc`
- Nainštaluj všetky balíky dostupné v *vyššej* (other) plain T_EX kolekcii:
`>> sh install-pkg.sh --collection=plain3`
- Umiestni všetky súbory, ktoré sú potrebné pre PStricks do tar súboru v `/tmp`:
`>> sh install-pkg.sh --package=pstricks \
>> --archive=/tmp/pstricks.tar`

3.4. Program `texconfig`

Po inštalácii, keď sú všetky súbory prekopírované na svoje miesta, môžete spustiť program nazvaný `texconfig`, ktorý vám umožňuje prispôsobiť systém vašim lokálnym potrebám. Tento program môžete spustiť aj kedykoľvek neskôr, keď budete potrebovať zmeniť vaše nastavenie a to buď v režime celej obrazovky (čo vyžaduje program `dialog`, zahrnutý v binárnych balíkoch) alebo v režime s príkazovým riadkom. Program by mal byť používaný pri každej údržbe, ako napr. zmena inštalovaných tlačiarň alebo prebudovanie databázy súborov. Oba režimy majú prístupnú nápovedu, ktorá vás prevedie cez ponúkané možnosti.

4. Inštalácia a použitie pod operačným systémom Windows

Táto sekcia sa vzťahuje iba na systémy Windows 9x alebo NT. Ak používate Windows 3.1, budete musieť nainštalovať `emTeX` z koreňového adresára `systems` ručne.

Takisto je potrebné, aby vaše Windows boli nastavené na používanie rozšírení Microsoft Joliet na čítanie CD-ROMov; pozrite si obsah CD-ROMu v Exploreri a zistite, či zobrazuje dlhé názvy súborov a rozlišuje veľké a malé znaky. Ak tomu tak nie je, nemôžete okamžite použiť systém tak, že ho spustíte z CD-ROMu.

Tento Win32 T_EX systém obsahuje nový prehliadač `dvi`, `Windvi`, ktorý je užívateľsky podobný zaužívanému `xdvi` v systéme Unix. Dokumentáciu k nemu nájdete na `texmf/doc/html/windvi/windvi.html`.

4.1. Spúšťanie z CD-ROMu

Všetky T_EX programy môžete spúšťať priamo z CD-ROMu, vrátane priameho prístupu ku všetkým makrám a fontom, avšak za cenu menšej výkonnosti

oproti inštalácii na pevnom disku. Pre efektívnu prácu potrebujete modifikovať premenné prostredia a vytvoriť nejaké malé pomocné adresáre na pevnom disku. Tieto adresáre budú obsahovať nutné konfiguračné súbory povoľujúce užívateľovi modifikovať nastavenia programov a generovať nutný formátový súbor. Navyše, automaticky generované fontové súbory tam budú tiež ukladané. Všetky tieto predbežné kroky sú vykonávané programom `TeXSetup.exe`, ktorý je volaný z adresáru `setupw32/` na CD-ROM. Ak program začal a pomocný adresár je vybraný, vyberte ‚Run CD‘ voľbu. Keď je inštalácia úplna, musíte reštartnúť Windows. Teraz môžete spúšťať programy z príkazového riadku alebo použitím \TeX ového editoru, ktorý spúšťa programy prostredníctvom pohodlnej ponuky.

4.2. Inštalácia na pevný disk

Inštalácia sa spúšťa priamo autoštartom CD alebo spustením programu `TeXsetup.exe` v adresári `setupw32`, ktorý pracuje s dostupnými zoznamami jednotlivých kolekcií a balíkov na CD-ROMe. Umožní vám zvoliť si stupeň inštalácie každej kolekcie (pozri časť 2.1) s bližším popisom kolekcií a balíkov, dovoľuje vám vynechať dokumentáciu a/alebo zdrojové segmenty balíkov, pokiaľ je váš diskový priestor obmedzený. Bude vám umožnené zvoliť si adresáre, do ktorých sa nainštaluje hlavná distribúcia a vaša lokálna konfigurácia. Navyše, budete mať možnosť si nainštalovať \TeX editor a prehliadač PostScriptových súborov Ghostscript.

Buďte si prosím vedomí toho, že veľkosť diskového ‚clustra‘ v partiáciách systému DOS môže radikálne ovplyvniť veľkosť vašej inštalácie \TeX u. Adresárový strom obsahuje stovky malých súborov a nie je nezvyčajné, že kompletná inštalácia zaberá až štvornásobne viac miesta, ako zaberala na CD-ROMe.

Keď je inštalácia dokončená, budete musieť reštartovať Windows a potom môžete spúšťať programy \TeX u z príkazového riadku alebo z menu ľubovoľného inštalovaného editoru.

Po prvej inštalácii nebežiacej z CD-Romu, budete mať možnosť pridať jednotlivé balíky k inštalácii. Ak to chceme urobiť, vyberte ‚Add TeX Package‘ voľbu z ‚TeX Live‘ → ‚Maintenance‘ systém menu.

Spustením `TeXSetup --help` budú zobrazené všetky dostupné voľby.

5. Budovanie na novej platforme Unixu

Ak máte platformu, pre ktorú sme neposkytli binárne súbory, budete si musieť sami skompilovať \TeX a príbuzné programy. Toto nie je až také ťažké, ako to znie. Všetko, čo potrebujete, je v adresári `source` na CD-ROMe.

Najprv budete musieť nainštalovať podporný adresárový strom z **TeX Live** CD-ROM (urobte základnú inštaláciu, bez zvolených systémových binárnych súborov).

5.1. Predpoklady

Na kompiláciu \TeX u a podporných programov budete potrebovať okolo 100 megabytov voľného diskového priestoru. Takisto budete potrebovať kompilátor ANSI C, program `make`, lexikálny analyzátor a parsovací generátor. GNU nástroje (`gcc`, GNU `make`, `m4`, `flex`, `bison`) sú najčastejšie testované na rôznych platformách. `gcc-2.7.* flex-2.4.7` a GNU `make-3.72.1` alebo novšie verzie by mali pracovať dobre. Môžete mať síce dobrú skúsenosť s prácou s inými kompilátormi C a programami `make`, ale aby ste si poradili s problémami je potrebné, aby ste dobre rozumeli aj stavbe programov Unixu. Príkaz `uname` musí vrátiť zmysluplnú hodnotu.

5.2. Konfigurácia

Najprv rozbaľte zdrojové súbory zo skompresovaného `tar` súboru v adresári `source` na váš disk a prejdite do adresáru, do ktorého ste ich umiestnili. Rozhodnite sa, kde bude mať byť umiestnený koreňový adresár inštalácie, napr. `/usr/local` alebo `/usr/local/TeX`. Budete zrejme musieť použiť ten istý adresár, do ktorého ste inštalovali podporný strom.

Teraz odštartujte proces konfigurácie spustením `configure` s príkazom

```
>> ./configure -prefix=/usr/local/TeX
```

„Prefixový“ adresár je ten, do ktorého ste nainštalovali podporný strom; rozloženie adresárov, ktoré sa použije je nasledovné (`$TEXDIR` je adresár, ktorý ste zvolili):

<code>\$TEXDIR/man</code>	manuálové stránky Unixu
<code>\$TEXDIR/share/texmf</code>	hlavný strom s fontami, makrami, atď
<code>\$TEXDIR/info</code>	manuály ku GNU štýlom
<code>\$TEXDIR/bin/\$PLATFORM</code>	binárne súbory

Ak chcete, môžete vynechať časť `,share/‘` pre adresár `texmf`, keďže `$TEXDIR/share/texmf` a `$TEXDIR/texmf` sa automaticky detekujú pri konfigurácii. Ak zvolíte niečo iné, budete musieť špecifikovať adresár možnosťou `--datadir` v `configure`.

Ak chcete vynechať úroveň adresáru `$PLATFORM` (t.j. umiestniť binárne súbory priamo do `$TEXDIR/bin`), použite možnosť `--disable-multiplatform` pre `configure`.

Ak sa chcete dozvedieť viac o ostatných možných nastaveniach pri konfigurácii (ako napr. nastavenie vynechania voliteľných balíkov ako Ω alebo $\varepsilon\text{-TeX}$), napíšte `./configure --help`.

5.3. Spúšťanie make

Uistite sa, že nie je nastavená premenná prostredia `noclobber` a napíšte

```
>> make world
```

a dajte si pauzu...

Mohlo by byť užitočné zaznamenať celý výstup, napr. napísaním

```
>> sh -c "make world >world.log 2>&1" &
```

Prv než si začnete myslieť, že všetko je v poriadku, skontrolujte prosím log súbor a presvedčte sa, či sa nevyskytli žiadne chyby (GNU make používa reťazec ,Error:' kedykoľvek príkaz vráti chybový kód) a skontrolujte, či boli vytvorené všetky binárne súbory:

```
>> cd /usr/local/TeX/bin/i686-pc-linux-gnu
```

```
>> ls | wc
```

Výsledok by mal byť 213.

Ak potrebujete pre make install špeciálne privilégia, môžete spustiť make dvakrát nezávisle za sebou:

```
>> make all
```

```
>> su
```

```
>> make install strip
```

5.4. Záverečné kroky konfigurácie

Pridajte do svojej PATH adresár obsahujúci práve nainštalované binárne súbory (napr. /usr/local/TeX/bin/mips-sgi-irix6.5); podobne pridajte do MANPATH a INFOPATH zodpovedajúce momentálne inštalované podadresáre, t.j. \$TEXDIR/man a \$TEXDIR/info.

Program texconfig vám umožňuje nastaviť požadované predefinované hodnoty pre delenie slov, veľkosť papiera, príkaz na tlač, METAFONT mód a pod. Tento program môžete buď spustiť interaktívne a pozrieť sa, aké možnosti ponúka alebo napísať

```
>> texconfig help
```

Ak napríklad nepoužívate formát papiera A4, môžete za predvolenú veľkosť papiera nastaviť ,lettersize' napísaním:

```
>> texconfig dvips paper letter
```

```
>> texconfig xdvi paper us
```

6. Používateľská príručka ku systému Web2c

Web2c obsahuje množinu T_EX-príbuzných programov, t.j. samotný T_EX, METAFONT, METAPOST, BIBT_EX, atď. Originálna implementácia pochádza od Tomáša Rokického, ktorý v roku 1987 vyvinul prvý T_EX-to-C systém adaptujúci zmenové súbory systému pre Unix, ktoré boli v prvom rade prácou Howarda Trickeya a Pavla Curtisa. Tim Morgan sa stal spravovateľom systému a počas jeho obdobia sa meno zmenilo na Web-to-C. V roku 1990 Karl Berry prebral

túto prácu, asistoval pri tuctoch dodatočných príspevkov a v roku 1997 podal taktovku Olafovi Weberovi. Posledným výsledkom je Web2c (verzia 7.3) z marca 1999, ktorý tvorí základ súčasného TeX Live CD-ROM.

Web2c 7.3 systém beží pod Unixom, Windows 3.1, 9x/NT, DOS a inými operačnými systémami. Používa originálne T_EX zdrojové súbory od Knutha a ostatné základné programy napísané vo `web`, ktoré sú preložené do C zdrojového kódu. Navyše, systém ponúka veľkú množinu makier a funkcií vyvinutých na rozšírenie originálneho T_EX software. Základné komponenty rodiny T_EXu sú:

<code>bibtex</code>	Spravovanie bibliografií.
<code>dmp</code>	Konverzia <code>troff</code> do MPX (METAPOST obrázky).
<code>dvicopy</code>	Vytvára modifikovanú kópiu DVI súboru.
<code>dvitomp</code>	Konverzia DVI do MPX (METAPOST obrázky).
<code>dvitype</code>	Konverzia DVI do ľudske-čitateľného textu.
<code>gftodvi</code>	Generovanie fontov pre náhľad.
<code>gftopk</code>	Konverzia <code>gf</code> formátu fontov do pakovaných fontov.
<code>gftype</code>	Konverzia <code>gf</code> formátu fontov do ľudske-čitateľného textu.
<code>makempx</code>	METAPOST značkové sádzanie.
<code>mf</code>	Vytváranie rodín fontov.
<code>mft</code>	Preddefinované METAFONTové zdrojové súbory.
<code>mpost</code>	Tvorba technických diagramov.
<code>mpto</code>	METAPOST značkový výber.
<code>newer</code>	Porovnanie modifikačných časov.
<code>patgen</code>	Vytváranie vzorov rozdeľovania slov.
<code>pktogf</code>	Konverzia pakovaných formátov fontov do <code>gf</code> formátov.
<code>pktype</code>	Konverzia pakovaných písiem do ľudske-čitateľného textu.
<code>pltotf</code>	Konverzia ‚Property list‘ do TFM.
<code>pooltype</code>	Zobrazovanie ‚web pool‘ súborov.
<code>tangle</code>	Konverzia <code>web</code> súborov do Pascalu.
<code>tex</code>	Sadzba.
<code>tftopl</code>	Konverzia TFM do ‚property list‘.
<code>vftovp</code>	Konverzia virtuálneho fontu do virtuálneho ‚property list‘.
<code>vptovf</code>	Konverzia virtuálneho ‚property list‘ do virtuálneho fontu.
<code>weave</code>	Konverzia <code>web</code> súborov do T _E Xu.

Presné funkcie a syntax týchto programov sú popísané v dokumentáciách jednotlivých balíkov alebo dokumentácii Web2c. Napriek tomu, poznanie niekoľkých princípov, ktoré platia pre celý balík programov vám pomôže vyťažiť čo najviac z vašej Web2c inštalácie.

Všetky programy dodržiavajú štandardné GNU voľby:

- `--help` Vypisuje prehľad základného používania.
- `--verbose` Vypisuje detailnú správu spracovania.
- `--version` Vypisuje informáciu o verzii, potom skončí.

Na vyhľadávanie súborov používajú Web2c programy prehľadávaciu knižnicu Kpathsea. Táto knižnica používa kombináciu premenných prostredia a niekoľkých konfiguračných súborov na optimalizáciu prehľadávania adresárového stromu \TeX u. Web2c 7.3 zvládne prácu s viacerými adresárovými stromami súčasne, čo je užitočné, keď niekto chce udržiavať štandardnú distribúciu \TeX u a jeho lokálne rozšírenia v dvoch rozličných stromoch. Na urýchlenie vyhľadávania súborov, koreň každého stromu obsahuje súbor `ls-R` so záznamom obsahujúcim meno a relatívnu cestu ku všetkým súborom umiestneným pod týmto koreňom.

6.1. Vyhľadávanie ciest knižnicou Kpathsea

Najprv popíšeme všeobecný mechanizmus vyhľadávania ciest knižnicou Kpathsea.

Vyhľadávacou cestou nazveme zoznam *elementov cesty*, ktorými sú v prvom rade mená adresárov oddelené dvojbodkou alebo bodkočiarkou. Vyhľadávacia cesta môže pochádzať z viacerých zdrojov. Pri vyhľadávaní súboru „my-file“ podľa cesty „./dir“, Kpathsea skontroluje každý element cesty: najprv `./my-file`, potom `/dir/my-file`, vracajúc prvý zodpovedajúci nájdený prvok (alebo prípadne všetky zodpovedajúce prvky).

Aby bolo dosiahnuté prispôsobenie sa konvenciám čo možno najviac operačných systémov, na neunixových systémoch Kpathsea môže používať oddeľovače názvov súborov rôzne od dvojbodky („:“) a lomítka („/“).

Pri kontrolovaní určitého elementu cesty p Kpathsea najprv overí, či sa na ňu nevzťahuje vopred vybudovaná databáza (pozri ‚Databáza názvov súborov‘ na strane 21), t.j., či sa databáza nachádza v adresári, ktorý je prefixom p . Ak tomu tak je, špecifikácia cesty sa porovnáva s obsahom databázy.

Ak databáza neexistuje, alebo sa nevzťahuje na tento element cesty, alebo sa v nej hľadaný súbor nevyskytuje, celý systém súborov je prehľadaný (pokiaľ to nebolo zakázané špecifikáciou začínajúcou „!!“ a hľadaný súbor musí existovať). Kpathsea zostrojí zoznam adresárov zodpovedajúcich tomuto elementu cesty a potom skontroluje každý z nich, či sa v ňom nenachádza hľadaný súbor.

Podmienka ‚súbor musí existovať‘ sa týka napr. súborov typu „.vf“ a vstupných súborov čítaných príkazom \TeX u `\openin`. Takéto súbory nemusia existovať (napr. `cmr10.vf`) a nebolo by dobré prehľadávať kvôli nim celý disk. Preto, keď zabudnete aktualizovať `ls-R` pri inštalácii nového „.vf“ súboru, súbor nebude nikdy nájdený. Každý element cesty sa prekontroluje — najprv databáza, potom disk. Keď je súbor nájdený, vyhľadávanie sa zastaví a výsledok je vrátený .

Hoci najjednoduchší a najbežnejší element cesty je meno adresáru, Kpathsea podporuje aj iné zdroje vo vyhľadávacích cestách: dedičné (layered) štandardné hodnoty, mená premenných prostredia, hodnoty súboru `config`, domáce adresáre

používateľov a rekurzívne prehľadávanie podadresárov. Preto, keď hovoríme, že Kpathsea *rozbalí* element cesty, znamená to, že pretransformuje všetky špecifikácie do základného mena alebo mien adresárov. Toto je popísané v nasledujúcich odsekoch.

Všimnite si, že keď je meno hľadaného súboru vyjadrené absolútne alebo explicitne relatívne, t.j. začína „/“ alebo „./“ alebo „../“, Kpathsea jednoducho skontroluje, či taký súbor existuje.

Zdroje cesty

Vyhľadávacia cesta môže byť vytvorená z rôznych zdrojov. Kpathsea ich používa v tomto poradí:

1. Používateľom nastavená premenná prostredia, napríklad `TEXINPUTS`. Premenné prostredia s pridanou bodkou a menom programu prepisujú momentálne nastavené; napríklad, keď „`latex`“ je meno práve bežiaceho programu, potom premenná `TEXINPUTS.latex` prepíše `TEXINPUTS`.
2. Programovo-špecifický konfiguračný súbor, napríklad riadok `,S /a:/b'` v súbore `config.ps dvips`.
3. Konfiguračný súbor Kpathsea — `texmf.cnf`, obsahujúci riadok ako: „`TEXINPUTS=/c:/d`“ (pozri ďalej).
4. Predvolené hodnoty počas kompilácie.

Všetky tieto hodnoty vyhľadávacej cesty môžete prezerat použitím debugovacích možností (pozri ‚Debugovanie‘ na strane 27).

Konfiguračné súbory

Kpathsea číta počas behu z *konfiguračných súborov* s menom `texmf.cnf` vyhľadávaciu cestu a ďalšie definície. Vyhľadávacia cesta používaná na hľadanie týchto súborov sa volá `TEXMFCNF` (v predvolenom nastavení sa tento súbor nachádza v podadresári `texmf/web2c`). *Všetky* súbory `texmf.cnf` vo vyhľadávacej ceste budú prečítané a definície v novších súboroch prepíšu definície v starších. Preto pri vyhľadávacej ceste `.$TEXMF`, hodnoty z `./texmf.cnf` prepíšu hodnoty z `$TEXMF/texmf.cnf`.

Pri čítaní popisu formátu súboru `texmf.cnf`, ktorý sa nachádza nižšie, pozrite si prosím aj prílohu 9, začínajúcu na strane 33, kde sa nachádza výpis súboru `texmf.cnf` z CD-ROMu.

- Komentáre začínajú znakom ‚%‘ a pokračujú do konca riadku.
- Prázdne riadky sú ignorované.
- Znak `\` na konci riadku slúži ako pokračovací znak, t.j. nasledujúci riadok je k nemu pripojený. Prázdne znaky na začiatku pripájaných riadkov nie sú ignorované.
- Všetky ostatné riadky majú tvar:

`variable[.programe] [=] value`

kde „=“ a prázdne znaky naokolo sú nepovinné.

- Meno premennej ‚*variable*‘ môže obsahovať akékoľvek znaky okrem prázdnych znakov, „=“, alebo „.“, ale obmedziť sa na znaky „A-Za-z-“ je najbezpečnejšie.
- Ak je ‚*.programe*‘ neprázdne, definícia sa použije iba vtedy, keď práve bežiaci program má meno *programe* alebo *programe.exe*. Toto umožňuje napríklad mať pre rôzne nadstavby T_EXu rôzne vyhľadávacie cesty.
- Hodnota ‚*value*‘ môže obsahovať akékoľvek znaky okrem ‚%‘ a ‚@‘. Na pravej strane nie je možné použiť ‚*\$var.prog*‘; namiesto toho musíte použiť ďalšiu premennú. Znak ‚;‘ vo ‚*value*‘ je preložený do ‚:‘ ak sme pod operačným systémom Unix. Toto je užitočné, keď chceme mať jediný `texmf.cnf` súbor pre systémy Unix, MSDOS a Windows.
- Všetky definície sú prečítané skôr, ako sa expandujú. Preto môžu existovať referencie na premenné skôr, ako sú tieto definované.

Ukážkový úsek konfiguračného súboru, ilustrujúci väčšinu týchto bodov nasleduje pod textom:

```
TEXMF                = {$TEXMFLOCAL;!!$TEXMFMAIN}
TEXINPUTS.latex      = .;$TEXMF/tex/{latex;generic;}//
TEXINPUTS.fontinst   = .;$TEXMF/tex//;$TEXMF/fonts/afm//
% e-TeX related files
TEXINPUTS.elatex     = .;$TEXMF/{etex;tex}/{latex;generic;}//
TEXINPUTS.etex       = .;$TEXMF/{etex;tex}/{eplain;plain;generic;}//
```

Expanzia cesty

Kpathsea rozpoznáva určité zvláštne znaky a konštrukcie vo vyhľadávacích cestách podobné tým, čo existujú v prostrediach Unixovských módov (shells). Ako všeobecný príklad uvidíme komplexnú cestu `~$USER/{foo,bar}//baz`, ktorá sa expanduje do všetkých podadresárov pod adresármi `foo` a `bar` v domovskom adresári používateľa `$USER`, ktorý obsahuje adresár alebo súbor `baz`. Tieto konštrukcie sú popísané v ďalších odsekoch.

Predvolená expanzia

Ak vyhľadávacia cesta s najväčšou prioritou (pozri ‚Zdroje cesty‘ na strane 18) obsahuje *dvojbodku navyše* (t.j. začiatočnú, koncovú, alebo zdvojenú), Kpathsea vloží na toto miesto vyhľadávaciu cestu s druhou najvyššou prioritou, ktorá je definovaná. Ak táto vložená cesta obsahuje dvojbodku navyše, to isté sa stane s ďalšou najvýznamnejšou cestou. Keby sme mali napríklad dané takéto nastavenie premennej prostredia

```
>> setenv TEXINPUTS /home/karl:
```

a hodnotu `TEXINPUTS` v súbore `texmf.cnf`

```
.:$TEXMF//tex
```

potom konečná hodnota použitá na vyhľadávanie by bola:

```
/home/karl.:$TEXMF//tex
```

Keďže by bolo zbytočné vkladať predvolenú hodnotu na viac ako jedno miesto, Kpathsea mení iba nadbytočnú „:“ a všetko ostatné ponecháva na mieste: kontroluje najprv začiatočnú „:“, potom koncovú „:“ a potom zdvojenú „:“.

Expanzia zátvoriek

Užitočná črta je expanzia zátvoriek, ktorá funguje tak, že napríklad `v{a,b}w` sa expanduje na `vaw:vw`. Vnárание je povolené. Toto môže byť použité na implementáciu viacnásobných T_EXovských hierarchií, priradením hodnoty `$TEXMF` s použitím zátvoriek. Napríklad v súbore `texmf.cnf` nájdete nasledujúcu definíciu:

```
TEXMF = {$HOMETEXMF,$TEXMFLOCAL,!!$VARTEXMF,!!$TEXMFMAIN}
```

Keď potom napíšete niečo podobné ako

```
TEXINPUTS = .;$TEXMF/tex//
```

bude to znamenať, že po hľadaní v aktuálnom adresári sa najprv prehľadá celý strom `$HOMETEXMF/tex`, `$TEXMFLOCAL/tex`, `$VARTEXMF/tex` a `$TEXMFMAIN/tex` (posledné dva s použitím databázových súborov `ls-R`). Je to vhodný spôsob ako spúšťať dve paralelné T_EX štruktúry, jednu nemennú (napríklad na CD-ROMe) a druhú neustále aktualizovanú novými verziami, akonáhle sú dostupné. Použitím premennej `$TEXMF` vo všetkých definíciách máme istotu, že sa vždy ako prvý prehľadá aktuálny strom.

Expanzia podadresárov

Dva alebo viac za sebou nasledujúcich znakov `/` v elemente cesty nasledujúcim za adresárom `d` je nahradený všetkými podadresármi `d`: najprv podadresármi priamo pod `d`, potom podadresármi pod nimi, atď. Poradie, v akom sú prehľadávané podadresáre na každej úrovni *nie je špecifikované*.

Ak po „//“, špecifikujete akékoľvek komponenty mena súboru, pridajú sa iba podadresáre so zodpovedajúcimi komponentami. Napríklad, „/a//b“ sa expanduje do adresárov `/a/1/b`, `/a/2/b`, `/a/1/1/b`, atď., ale nie do `/a/b/c` alebo `/a/1`.

Viacnásobné „//“ konštrukcie v ceste sú možné, ale „//“ na začiatku cesty je ignorované.

Typy špeciálnych znakov a ich význam: zhrnutie

Nasledujúci zoznam zahŕňa význam zvláštnych znakov v konfiguračných súboroch Kpathsea.

: Oddeľovač v špecifikácii cesty; na začiatku alebo na konci cesty nahrádza predvolenú expanziu cesty.

```

;   Oddeľovač v neunixových systémoch (správa sa ako :).
$   Expanzia premennej.
~   Reprezentuje domovský adresár používateľa.
{..} Expanzia zátvoriek, napr. z a{1,2}b sa stane a1b:a2b.
//  Expanzia podadresárov. (Môže sa vyskytnúť kdekoľvek v ceste, okrem jej
    začiatku).
%   Začiatok komentáru.
\   Znak pokračovania riadku (umožňuje viacriadkové vstupy).
!!  Povel na hľadanie súboru iba v databáze, neprehľadáva disk.

```

6.2. Súborové databázy

Kpathsea minimalizuje prístupy na disk pri vyhľadávaní. Predsa však pri inštaláciách s dostatočným množstvom adresárov hľadanie súboru v každom možnom adresári môže zabráť prehnane veľa času (toto platí zvlášť vtedy, keď musia byť prejdené stovky adresárov s fontami). Kpathsea preto používa externe vytvorený „databázový“ súbor nazývaný `ls-R`, ktorý mapuje súbory v adresároch a pomáha tak vyhnúť sa vyčerpujúcejmu prehľadávaniu disku.

Skratky mien (`aliases`) v druhom databázovom súbore vám umožňujú dať dodatočné mená súborom nachádzajúcim sa v zozname `ls-R`. Toto môže byť užitočné pri prispôbovaní sa „8.3“-súborovým konvenciám DOSu v zdrojových súboroch.

Súborová databáza

Ako bolo vysvetlené hore, meno hlavnej databázy súborov musí byť `ls-R`. Môžete umiestniť jednu do koreňa každej hierarchie \TeX u vo vašej inštalácii ktorú chcete, aby bola prehľadávaná (predvolená je `$TEXMF`); väčšinou sa jedná iba o jednu hierarchiu. Kpathsea hľadá `ls-R` súbory podľa cesty v `TEXMFDBS`.

Odporúčaný spôsob ako vytvoriť a udržiavať „`ls-R`“ je spustiť skript `mktexlsr` zahrnutý v distribúcii. Je vyvolávaný rôznymi „`mktex`“... skriptami. Tento skript v princípe iba spúšťa príkaz

```
cd /your/teamp/root && ls -LAR ./ >ls-R
```

predpokladajúc, že `ls` vášho systému vytvára správny výstup (výstup GNU `ls` je v poriadku). Aby ste sa ubezpečili, že databáza bude vždy aktuálna, najjednoduchšie je pravidelne ju prebudovávať cez `cron`, takže po zmenách v inštalovaných súboroch — napríklad pri inštalácii alebo aktualizácii balíka \LaTeX u bude súbor `ls-R` automaticky aktualizovaný.

Ak súbor nie je v databáze nájdený, podľa predvoleného nastavenia Kpathsea začne vyhľadávať na disku. Ak však určitý element cesty začína „!`!`“, bude prehľadávaná *iba* databáza, nikdy nie disk.

kpsewhich: Samostatné prehľadávanie cesty

Program `kpsewhich` vykonáva prehľadávanie cesty nezávislé od každej aplikácie. Môže byť užitočný ako vyhľadávaci `find` program na nájdenie súborov v hierarchiách \TeX U (veľmi sa využíva v distribuovaných „`mktext`“... skriptoch).

>> `kpsewhich option... filename...`

Volby špecifikované v `,option‘` môžu začínať buď „-“ alebo „--“ a každá skratka, ktorá nie je viacznačná, je akceptovaná.

Kpathsea považuje každý element vstupného riadku, ktorý nie je argumentom nejakej volby za meno súboru, ktorý hľadá a vracia prvý súbor, ktorý nájde. Neexistuje voľba umožňujúca vrátiť všetky súbory s určitým menom (na to môžete použiť nástroj Unixu „`find`“).

Ďalšie dôležitejšie volby sú popísané nižšie.

`--dpi=num` Nastav rozlíšenie na `,num‘`; toto má vplyv iba na „`gf`“ a „`pk`“ vyhľadávanie. „-D“ je synonymom, kvôli kompatibilite s `dvips`. Predvolená hodnota je 600.

`--format=name`

Nastav formát pre vyhľadávanie na `,name‘`. Podľa predvoleného nastavenia je formát uhádnutý z mena súboru. Pre formáty, ktoré nemajú asociovanú jednoznačnú príponu, ako napríklad podporné súbory `METAPOSTu` a konfiguračné súbory `dvips`, musíte špecifikovať meno nájdené v prvom stĺpci Tabuľky 1, v ktorej je zoznam rozpoznávaných mien, popis, asociované premenné prostredia¹, a možné prípony súborov.

Tabuľka 1: Súborové typy Kpathsea

<i>Meno</i>	<i>Popis</i>	<i>Premenné</i>	<i>Prípony</i>
<code>afm</code>	Metriky písom Adobe	<code>AFMFonts</code>	<code>.afm</code>
<code>base</code>	Výpis pamäti <code>METAFontu</code>	<code>MFBASES</code> , <code>TEXMFINI</code>	<code>.base</code>
<code>bib</code>	Zdrojové súbory <code>BIBTeXu</code>	<code>BIBINPUTS</code> , <code>TEXBIB</code>	<code>.bib</code>
<code>bst</code>	Súbory štýlov <code>BIBTeX</code>	<code>BSTINPUTS</code>	<code>.bst</code>
<code>cnf</code>	Konfiguračné súbory čítané za behu	<code>TEXMFCNF</code>	<code>.cnf</code>
<code>dvips config</code>	Konfiguračné súbory <code>dvips</code> , napr., <code>config.ps</code> a <code>psfonts.map</code>	<code>TEXCONFIG</code>	<code>.map</code>
<code>fnt</code>	Predkompilované formáty \TeX U	<code>TEXFORMATS</code> , <code>TEXMFINI</code>	<code>.fnt</code> , <code>.efnt</code> , <code>.efm</code>

¹Definíciu týchto premenných prostredia môžete nájsť v súbore `texmf.cnf` (na strane 33).

Súborové typy Kpathsea *pokračovanie*

<i>Meno</i>	<i>Popis</i>	<i>Premenné</i>	<i>Prípony</i>
gf	Bitmapa generického fondu	FONTS, GFFONTS, GLYPHFONTS, TEXFONTS	.gf
graphic/figure	Zapúzdrené PostScript obrázky	TEXPICTS, TEXINPUTS	.eps, .epsi
ist	Súbory štýlov makeindex	TEXINDEXSTYLE, INDEXSTYLE	.ist
ls-R	Súborové databázy	TEXMFDBS	
map	Mapy písem	TEXFONTMAPS	.map
mem	Predkompilované formáty METAPOSTu	MPMEMS, TEXMFINI	.mem
mf	Zdrojové súbory METAFONT	MFINPUTS	.mf
mfpool	Programové súboru k METAFONTu	MFPOOL, TEXMFINI	.pool
mft	Súbor štýlov MFT	MFTINPUTS	.mft
mp	Zdrojové súbory METAPOSTu	MPINPUTS	.mp
mppool	Programové súbory k METAFONTu	MPPPOOL, TEXMFINI	.pool
METAPOST support	Podporné súbory pre METAPOST, používané DMP	MPSUPPORT	
ocp	Skompilované Ω súbory	OCPINPUTS	.ocp
ofm	Metriky písíem Ω	OFMFONTS, TEXFONTS	.ofm, .tfm
opl	Zoznamy vlastností Ω	OPLFONTS, TEXFONTS	.opl
otp	Translačné procesné Ω súbory	OTPINPUTS	.otp
ovf	Virtuálne fonty Ω	OVPFONTS, TEXFONTS	.ovf
ovp	Virtuálne zoznamy vlastností Ω	OVPFONTS, TEXFONTS	.ovp
pk	spakované bitmapové fonty	<i>program</i> FONTS (<i>program</i> being XDVl, etc.), PKFONTS, TEXPKS, GLYPHFONTS, TEXFONTS	.pk
PostScript header	Preddefionvané PostScriptové headre	TEXPSHEADERS, PSHEADERS	.pro, .enc
tex	Zdrojový súbor \TeX u	TEXINPUTS	.tex, .cls, .sty, .clo, .def
TeX system documentation	Súborová dokumentácia pre systém \TeX	TEXDOCS	

Súborové typy Kpathsea *pokračovanie*

<i>Meno</i>	<i>Popis</i>	<i>Premenné</i>	<i>Prípony</i>
TeX system sources	Zdrojové súbory pre systém T _E X	TEXSOURCES	
texpool	Programové súbory k T _E Xu	TEXPOOL, TEXMFINI	.pool
tfm	Metriky písiem T _E Xu	TFMFonts, TEXFonts	.tfm
Troff fonts	Fonty Troff, používané DMP	TRFonts	
truetype fonts	Obrysové fonty TrueType	TTFonts	.ttf, .ttc
type1 fonts	Obrysové fonty Type 1 PostScript	T1Fonts, T1Inputs, TEXPSHeaders, DVIPSHeaders	.pfa, .pfb
type42 fonts	Obrysové fonty Type 42 PostScript	T42Fonts	
vf	Viruálne fonty	VFFonts, TEXFonts	.vf
web2c files	Podporné súbory Web2c	WEB2C	
other text files	textové súbory používané ‚foo‘	FOOInputs	
other binary files	binárne súbory používané ‚foo‘	FOOInputs	

Posledné dve položky v Tabuľke 1 sú špeciálne prípady, kedy cesta a premenné prostredia závisia na mene programu: meno premennej sa vytvorí tak, že meno programu prepíšeme veľkými písmenami a pridáme INPUTS.

Premenné prostredia sa obyčajne nastavujú z konfiguračného súboru `texmf.cnf`. Explicitne ich nastavujte pri spúšťaní jedine vtedy, keď chcete prepísať jednu alebo viac hodnôt špecifikovaných v tomto súbore.

Všimnite si, že voľby „`--format`“ a „`--path`“ sa vzájomne vylučujú.

`--mode=string`

Nastav meno módu na `string`; toto má vplyv iba na „`gf`“ a „`pk`“ vyhľadávanie. Žiadna predvolená hodnota: každý mód bude nájdený.

`--must-exist`

Urob všetko preto, aby si našiel súbory. Ak je to potrebné, vrátane hľadania na disku. Normálne je v záujme efektívnosti prehľadávaná iba databáza `ls-R`.

`--path=string`

Vyhľadávaj podľa cesty `string` (oddeľovaná dvojbodkou ako zvyčajne) namiesto hádania vyhľadávacej cesty z mena súboru. Podporované sú „`///`“ a všetky bežné expanzie. Voľby „`--path`“ a „`--format`“ sa vzájomne vylučujú.

`--progrname=name`

Nastav meno programu na *,name‘*. Toto nastavenie ovplyvňuje použitie vyhľadávacej cesty cez nastavenie *,.progrname‘* v konfiguračných súboroch. Predvolená hodnota je „*kpsewhich*“.

`--show-path=name`

Zobrazí cestu použitú na vyhľadávanie súboru s typom *,name‘*. Môže byť použitá buď súborová prípona („*.pk*“, „*.vf*“ a pod.) alebo meno, podobne ako vo voľbe „`--format`“.

`--debug=num`

Nastaví počet debugovacích možností na *,num‘*.

Príklady použitia

Pozrime sa na Kpathsea v akcii.

```
>> kpsewhich article.cls
```

```
/usr/local/texmf/tex/latex/base/article.cls
```

Hľadáme súbor `article.cls`. Keďže prípona „`.cls`“ je jednoznačná, nemusíme špecifikovať, že hľadáme súbor typu `,tex‘` (zdrojový súbor $\text{T}_{\text{E}}\text{X}$). Nájdem ho v podadresári `tex/latex/base` pod koreňovým adresárom „`TEXMF`“. Podobne, všetky nasledujúce súbory budú nájdené bez problémov vďaka ich jednoznačnej príponě.

```
>> kpsewhich array.sty
```

```
/usr/local/texmf/tex/latex/tools/array.sty
```

```
>> kpsewhich latin1.def
```

```
/usr/local/texmf/tex/latex/base/latin1.def
```

```
>> kpsewhich size10.clo
```

```
/usr/local/texmf/tex/latex/base/size10.clo
```

```
>> kpsewhich small2e.tex
```

```
/usr/local/texmf/tex/latex/base/small2e.tex
```

```
>> kpsewhich tugboat.bib
```

```
/usr/local/texmf/bibtex/bib/beebe/tugboat.bib
```

Posledným súborom je bibliografická databáza `BIB $\text{T}_{\text{E}}\text{X}$` u pre články *TUG-Boatu*.

```
>> kpsewhich cmr10.pk
```

Bitmapové súbory fontov typu `.pk` sa používajú zobrazovacími programami ako `dvips` a `xdvi`. V tomto prípade je vrátený prázdny výsledok, keďže neexistujú žiadne vopred generované Computer Modern „`.pk`“ súbory v našom systéme (vzhľadom na to, že používame verzie `Type1` na `CD-ROMe`).

```
>> kpsewhich ecrm1000.pk
```

```
/usr/local/texmf/fonts/pk/ljfour/jknappen/ec/ecrm1000.600pk
```

Kvôli rozšíreným Computer Modern súborom sme museli vygenerovať „`.pk`“ súbory. Keďže predvolený mód `METAFONT` v našej inštalácii je `ljfour so`

základným rozlíšením 600 dpi (dots per inch), je vrátená táto inštancia.

```
>> kpsewhich -dpi=300 ecrm1000.pk
```

V tomto prípade po špecifikovaní, že nás zaujíma rozlíšenie 300dpi (`-dpi=300`) vidíme, že taký font nie je v systéme k dispozícii. Program ako `dvips` alebo `xdvi` by v tomto prípade vytvorili `.pk` súbory požadovaného rozlíšenia, použijúc skript `mktexpk`.

Teraz obráťme našu pozornosť na hlavičkové a konfiguračné súbory `dvips`. Najprv sa pozrieme na jeden z bežne používaných súborov, všeobecný prológový `tex.pro` na podporu \TeX u, potom pohladáme konfiguračný súbor (`config.ps`) a PostScriptovú mapu fontov `psfonts.map`. Keďže prípona „.ps“ je nejednoznačná, musíme pre súbor `config.ps` špecifikovať explicitne, o ktorý typ sa zaujímate (`,dvips config‘`).

```
>> kpsewhich tex.pro
```

```
/usr/local/texmf/dvips/base/tex.pro
```

```
>> kpsewhich --format='dvips config' config.ps
```

```
/usr/local/texmf/config/config.ps
```

```
>> kpsewhich psfonts.map
```

```
/usr/local/texmf/dvips/base/psfonts.map
```

Teraz sa pozrieme na podporné súbory URW Times PostScript. V Berryho schéme meno pre tieto pomenovania fontov je „`utm`“. Prvý súbor, ktorý hľadáme, je konfiguračný súbor, ktorý obsahuje meno mapového súboru:

```
>> kpsewhich --format="dvips config" config.utm
```

```
/usr/local/texmf/dvips/psnfss/config.utm
```

Obsah tohoto súboru je

```
p +utm.map
```

čo odkazuje na súbor `utm.map`, ktorý ideme ďalej hľadať.

```
>> kpsewhich --format="dvips config" utm.map
```

```
/usr/local/texmf/dvips/psnfss/utm.map
```

Tento mapový súbor definuje mená súborov fontov typu Type1 PostScript v kolekcii URW. Jeho obsah vyzerá takto (zobrazili sme iba jeho časť):

```
utmb8r NimbusRomNo9L-Medi ... <utmb8a.pfb
utmbi8r NimbusRomNo9L-MediItal... <utmbi8a.pfb
utmr8r NimbusRomNo9L-Regu ... <utmr8a.pfb
utmri8r NimbusRomNo9L-ReguItal... <utmri8a.pfb
utmbo8r NimbusRomNo9L-Medi ... <utmb8a.pfb
utmro8r NimbusRomNo9L-Regu ... <utmr8a.pfb
```

Zoberme napríklad, inštanciu Times Regular `utmr8a.pfb` a nájdime jej pozíciu v adresárovom strome `texmf` použitím vyhľadávania fontových súborov Type1:

```
>> kpsewhich utmr8a.pfb
```

```
/usr/local/texmf/fonts/type1/urw/utm/utmr8a.pfb
```

Z týchto príkladov by malo byť zrejmé, ako ľahko môžete nájsť umiestnenie daného súboru. Toto je zvlášť dôležité keď máte podozrenie, že ste narazili na

zlú verziu súboru, keďže `kpsewhich` vám zobrazí prvý súbor, ktorý zodpovedá vašim požiadavkam.

Debugovanie

Niekedy je potrebné vyšetriť ako program rozpoznáva referencie na súbory. Aby toto bolo možné vhodne uskutočniť, `Kpathsea` ponúka rôzne stupne debugovania:

- 1 Volania `stat` (testy súborov). Pri behu s aktuálnou `ls-R` databázou by nemal dať takmer žiaden výstup.
- 2 Referencie do hašovacích tabuliek (ako `ls-R` databáza, mapové súbory, konfiguračné súbory).
- 4 Operácie otvárania a zatvárania súboru.
- 8 Všeobecná informácia o ceste pre typy súborov hľadaných `Kpathsea`. Toto je užitočné pri zisťovaní, kde bola definovaná určitá cesta pre daný súbor.
- 16 Adresárový zoznam pre každý element cesty (vzťahuje sa iba na vyhľadávanie na disku).
- 32 Vyhľadávanie súborov.

Hodnota `-1` nastaví všetky horeuvedené voľby, v praxi pravdepodobne vždy použijete tieto úrovne ak budete potrebovať akékoľvek debugovanie.

Podobne s programom `dvips` nastavením kombinácie debugovacích prepínačov môžete detailne sledovať, odkiaľ sa berú používané súbory. Alternatívne, keď súbor nie je nájdený, debugovacia cesta ukazuje, v ktorých adresároch program daný súbor hľadal, čo môže naznačovať, v čom sa asi vyskytol problém.

Všeobecne povedané, keďže väčšina programov volá knižnicu `Kpathsea` vnútorne, debugovacie voľby je možné nastaviť pomocou premennej prostredia `KPATHSEA_DEBUG` na potrebnú kombináciu, ako je to popísané v horeuvedenom zozname.

(Poznámka pre používateľov Windows: nie je jednoduché presmerovať všetky hlášky v tomto systéme do súboru. Pre diagnostikovacie účely môžete dočasne priradiť

```
SET KPATHSEA_DEBUG_OUTPUT=err.log).
```

Uvažujme ako príklad malý zdrojový súbor `!ATEXu, hello-world.tex`, ktorý obsahuje nasledujúci vstup.

```
\documentclass{article}
\begin{document}
Hello World!
\end{document}
```

Tento malý súbor používa iba font `cmr10`, takže pozrime sa, ako `dvips` pripravuje PostScriptový súbor (chceme použiť `Type1` verziu písem `Computer Modern`, preto je nastavená voľba `-Pcms`).

```
>> dvips -d4100 hello-world -Pcms -o
```

V tomto prípade sme skombinovali dvips debugovaciu triedu 4 (cesty k fontom) s expanziou elementu cesty Kpathsea (pozri Referenčný Manuál dvips, texmf/doc/html/dvips/dvips_toc.html). Výstup, trochu preusporiadaný, je zobrazený na obrázku 4.

dvips začne lokáciou svojich pracovných súborov. Najprv je nájdený `texmf.cnf` ktorý obsahuje definície vyhľadávacích ciest ostatných súborov, potom databáza súborov `ls-R` (na optimalizáciu vyhľadávania súborov) a skratky mien súborov (`aliases`), čo robí možným deklarovať viacero mien (napr. krátke meno typu `,8.3'` ako v DOSe a viac prirodzenú dlhšiu verziu) pre ten istý súbor. Potom dvips pokračuje v hľadaní všeobecného konfiguračného súboru `config.ps` skôr, ako začne hľadať súbor nastavení `.dvipsrc` (ktorý, v tomto prípade, *nie je nájdený*). Nakoniec, dvips nájde konfiguračný súbor pre font Computer Modern PostScript, `config.cms` (toto bolo iniciované voľbou `-Pcms` v príkaze `dvips`). Tento súbor obsahuje zoznam „mapových“ súborov, ktoré definujú vzťah medzi menami fontov v \TeX u, PostScripte a systéme súborov.

```
>> more /usr/local/texmf/dvips/cms/config.cms
```

```
p +ams.map
p +cms.map
p +cmbkm.map
p +amsbkm.map
```

dvips preto pokračuje v hľadaní všetkých týchto súborov plus všeobecného mapového súboru `psfonts.map`, ktorý sa načítava vždy (obsahuje deklarácie bežne používaných PostScriptových fontov; pozri poslednú časť sekcie 6.2 kde sa nachádza viac detailov o narábaní s mapovými súborami PostScriptu).

V tomto bode sa dvips identifikuje používateľovi...

```
This is dvips 5.78 Copyright 1998 Radical Eye Software
(www.radicaleye.com)
```

...potom pokračuje v hľadaní prológového súboru `texc.pro`:

```
kdebug:start search(file=texc.pro, must_exist=0, find_all=0,
  path=./tex/dvips//:!!/usr/local/texmf/dvips//:
  ~/tex/fonts/type1//:!!/usr/local/texmf/fonts/type1//).
kdebug:search(texc.pro) => /usr/local/texmf/dvips/base/texc.pro
```

Po nájdení tohoto súboru, dvips napíše na výstup dátum a čas a informuje nás, že vygeneruje súbor `hello-world.ps`, že potrebuje súbor s fontom `cmr10`, ktorý bude deklarovaný ako „rezidentný“:

```
TeX output 1998.02.26:1204' -> hello-world.ps
```

```
Defining font () cmr10 at 10.0pt
```

```
Font cmr10 <CMR10> is resident.
```

Teraz sa rozbehne hľadanie súboru `cmr10.tfm`, ktorý je nájdený, potom je referencovaných ešte niekoľko prológových súborov (nezobrazené) a nakoniec je nájdená inštancia fontu `Type1`, `cmr10.pfb`, ktorá je pridaná do výstupného súboru (pozri posledný riadok).

```

debug:start search(file=texmf.cnf, must_exist=1, find_all=1,
  path=./usr/local/bin/texlive:/usr/local/bin:
    /usr/local/bin/texmf/web2c:/usr/local:
    /usr/local/texmf/web2c/././teTeX/TeX/texmf/web2c:).
kdebug:start search(file=ls-R, must_exist=1, find_all=1,
  path=~/.tex:/usr/local/texmf).
kdebug:search(ls-R) => /usr/local/texmf/ls-R
kdebug:start search(file=aliases, must_exist=1, find_all=1,
  path=~/.tex:/usr/local/texmf).
kdebug:search(aliases) => /usr/local/texmf/aliases
kdebug:start search(file=config.ps, must_exist=0, find_all=0,
  path=./~/tex:!!/usr/local/texmf/dvips//).
kdebug:search(config.ps) => /usr/local/texmf/dvips/config/config.ps
kdebug:start search(file=/root/.dvipsrc, must_exist=0, find_all=0,
  path=./~/tex:!!/usr/local/texmf/dvips//).
search(file=/home/goossens/.dvipsrc, must_exist=1, find_all=0,
  path=./~/tex/dvips//:!!/usr/local/texmf/dvips//).
kdebug:search($HOME/.dvipsrc) =>
kdebug:start search(file=config.cms, must_exist=0, find_all=0,
  path=./~/tex/dvips//:!!/usr/local/texmf/dvips//).
kdebug:search(config.cms)
=>/usr/local/texmf/dvips/cms/config.cms

```

Obr. 4: Hľadanie konfiguračných súborov

```

kdebug:start search(file=texc.pro, must_exist=0, find_all=0,
  path=./~/tex/dvips//:!!/usr/local/texmf/dvips//:
    ~/tex/fonts/type1//:!!/usr/local/texmf/fonts/type1//).
kdebug:search(texc.pro) => /usr/local/texmf/dvips/base/texc.pro

```

Obr. 5: Hľadanie prológového súboru

```

kdebug:start search(file=cmr10.tfm, must_exist=1, find_all=0,
  path=./~/tex/fonts/tfm//:!!/usr/local/texmf/fonts/tfm//:
    /var/tex/fonts/tfm//).
kdebug:search(cmr10.tfm)
=> /usr/local/texmf/fonts/tfm/public/cm/cmr10.tfm
kdebug:start search(file=texps.pro, must_exist=0, find_all=0,
  ...
<texps.pro>
kdebug:start search(file=cmr10.pfb, must_exist=0, find_all=0,
  path=./~/tex/dvips//:!!/usr/local/texmf/dvips//:
    ~/tex/fonts/type1//:!!/usr/local/texmf/fonts/type1//).
kdebug:search(cmr10.pfb)
=> /usr/local/texmf/fonts/type1/public/cm/cmr10.pfb
<cmr10.pfb>[1]

```

Obr. 6: Hľadanie fontového súboru

\TeX u, tieto parametre počas behu programu poskytujú praktický kompromis, ktorý dovoľuje aspoň nejakú flexibilitu.

7. História a poďakovania

Táto CD-ROM distribúcia je spojeným úsilím \TeX Users Group (Združenie používateľov \TeX u), britskej \TeX Users Group, francúzskej \TeX Users (GUTenberg) a nemeckej \TeX Users (DANTE e.V.) s podporou česko-slovenských, holandských, indických a poľských združení užívateľov. Diskusia začala na sklonku roku 1993, keď holandská \TeX Users Group začínala prácu na svojom 4All \TeX CD-ROM pre používateľov MS-DOSu a vtedy vznikla nádej na vydanie jedného, racionálneho CD-ROM pre všetky systémy. Toto bol veľmi ambiciózný cieľ, ktorý ale splodil nielen veľmi úspešný CD-ROM 4All \TeX u, ale aj TUG Technical Council pracovnú skupinu zaoberajúcu sa *\TeX Directory Structure*, ktorá špecifikovala, ako vytvoriť konzistentné a zvládnuteľné kolekcie podporných súborov \TeX u. Finálny koncept TDS bol publikovaný v decembri roku 1995 vo vydaní *TUGboat* a od prvých štádií bolo jasné, že jedným žiadúcim produktom by mohla byť modelová štruktúra na CD-ROM. CD-ROM, ktorý práve máte, je priamym výsledkom rokovaní pracovnej skupiny. Úspech CD-ROM 4All \TeX u takisto ukázal, že aj používatelia Unixu by radi ťažili z podobne jednoduchého systému a toto je ďalšou hlavnou nitkou **\TeX Live**.

Na jeseň 1995 sme sa odhodlali urobiť nový CD-ROM obsahujúci TDS pre Unix a čoskoro sme identifikovali te \TeX Thomasa Essera ako ideálne nastavenie, keďže už mal podporu pre viacero platforiem a bol vybudovaný s ohľadom na prenositeľnosť medzi rôznymi správami súborov. Thomas súhlasil s pomocou a serióznou spolupráca začala na začiatku roku 1996. Prvé vydanie prišlo na svet v máji 1996. Na začiatku roku 1997, Karl Berry dokončil hlavné vydanie jeho Web2c balíka, ktoré obsahovalo takmer všetky črty, ktoré Thomas Esser pridal do te \TeX u a rozhodli sme sa založiť druhé vydanie CD-ROM na štandarde Web2c s pridaním skriptu `texconfig` prebratého z te \TeX u. Tretie vydanie CD-ROM bolo založené na hlavnej revízii Web2c 7.2, ktorú uskutočnil Olaf Weber; v tom istom čase sa pracovalo na novej revízii te \TeX u a **\TeX Live** obsahuje už takmer všetky svoje črty. Štvrté vydanie prebiehalo podľa takého istého vzoru, použijúc novú verziu te \TeX u a nové vydanie Web2c (7.3).

V piatom vydaní (apríl 2000) veľa častí CD-ROM bolo revidovaných a skontrolovaných, stovky balíkov bolo nahradených novšími verziami. Omega a pdf \TeX sú v nových revidovaných vydaniach. Časť \TeX ových podporných programov (napr. `xdvi`, `dvips` a `tex4ht`) bola tiež revidovaná.

Najväčšia zmena v \TeX Live 5 je, že všetok softvér, ktorý nie je voľne šíriteľný, nebol do CD-ROM zaradený. Všetko na tomto CD-ROM by malo

byť kompatibilné s ‚Debian Free Software Guidelines‘ (<http://www.debian.org/intro/free>). Urobili sme, ako sme najlepšie vedeli, kontrolu licenčných podmienok pre všetky balíky. Budeme veľmi povďačný za oznámenie každej chyby.

Zvlášť by sme chceli poďakovať:

- Nemeckej T_EX Users (DANTE e.V.), ktorá poskytla počítač, na ktorom obsah CD-ROMu bol vyvinutý a udržiavaný; Rainer Schöpferovi a Reinhard Zierkemu, ktorí nato dozerali;
- Perforce company, ktorá poskytla voľnú kópiu vynikajúceho zmenového manažmentového systému, ktorú sme používali na manažovanie obsahu CD-ROMu;
- Karl Berrymu, ktorý poskytol originálnu distribúciu Web2c a pokračoval s poskytovaním hodnotných rád, povzbudením a pomocou;
- Mimi Burbank, ktorá zabezpečila prístup na Florida State University Supercomputer Research Institute k rôznym počítačom, na ktorých bol kompilovaný T_EX a pomohla nám kedykoľvek to bolo treba;
- Kaja Christiansen, ktorá poskytla podstatnú spätnú väzbu, kompilácie a pripravovala dokumentáciu;
- Thomas Esserovi, bez jeho obdivuhodného balíku teT_EX by tento CD-ROM pravdepodobne neexistoval a jeho ustavičná pomoc robí z tohto CD-ROM neustále lepší produkt;
- Eitan Gurariovi, ktorého T_EX4ht bol použitý na vytvorenie HTML verzie tejto dokumentácie a ktorý neúnavne pracoval, aby ju vylepšil v každom ohľade;
- Artu Ogawovi a Pat Monohon, ktorí koordinovali tento release pre TUG;
- Petrovi Olšákovi, ktorý pozorne koordinoval a monitoroval všetky česko-slovenské materiály;
- Fabrice Popineau, ktorý nepretržite pracoval na Win32 časti balíka (hlavne setupu) a prispel v mnohých smeroch svojimi myšlienkami, radami a kódom;
- Staszek Wawrykiewiczovi, ktorý poskytoval spätnú väzbu vo veľkej miere a koordinoval poľské príspevky;
- Olaf Weberovi za jeho trpezlivé vytvorenie a spravovanie Web2c 7.3;
- Graham Williamsovi, na ktorého práci stojí katalóg balíkov.

Alain Rabaute, Pascal Quignon, Gerhard Wilhelms, Fabrice Popineau, Janka Chlebiková, Staszek Wawrykiewicz, Erik Frambach a Ulrik Vieth láskavo preložili dokumentáciu do svojich jazykov, skontrolovali ostatnú dokumentáciu a poskytli veľmi vítanú spätnú väzbu.

8. Budúce verzie

Tento CD-ROM nie je dokonalým produktom! Plánujeme ho opätovne vydávať raz za rok a radi by sme poskytli viac nápovery, viac nástrojov, viac inštalačných programov a (samozrejme), neustále vylepšovaný a kontrolovaný strom makier a fontov. Všetka táto práca je dielom dobrovoľníkov pod veľkým tlakom ich obmedzeného voľného času. Napriek tomu jej ešte ostáva urobiť veľký kus. Ak môžete s touto prácou pomôcť, neváhajte a začnite!

Korekcie, návrhy a príspevky do budúcich revízií môžete posilať na adresu:

Sebastian Rahtz
7 Stratfield Road
Oxford OX2 7BG
United Kingdom
rahtz@tug.org

Aktuality, poznámky a návrhy budú uvedené na CTAN v `info/texlive`. WWW stránka s informáciami a detailami ohľadom objednávanía sa nachádza na adrese <http://www.tug.org/tex-live.html>.

Komentáre k česko-slovenským \TeX ovým veciam sú vítané na adresách `petr.olsak@cstug.cz` a/alebo `jana.chlebikova@cstug.cz`.

9. Súbor `texmf.cnf`

```
2 % TeX Live texmf.cnf
3 % What follows is a super-summary of what this .cnf file can
4 % contain. Please read the Kpathsea manual for more information.
5 %
6 % texmf.cnf is generated from texmf.in, by replacing @var@ with the
7 % value of the Make variable 'var', via a sed file texmf.sed, generated
8 % (once) by kpathsea/Makefile (itself generated from kpathsea/Makefile.in
9 % by configure).
10 %
11 % Any identifier (sticking to A-Za-z_ for names is safest) can be assigned.
12 % The '=' (and surrounding spaces) is optional.
13 % No % or @ in texmf.in, for the sake of autogeneration.
14 % (However, %'s and @'s can be edited into texmf.cnf or put in envvar values.)
15 % $foo (or ${foo}) in a value expands to the envvar or cnf value of foo.
16 %
17 % Earlier entries (in the same or another file) override later ones, and
18 % an environment variable foo overrides any texmf.cnf definition of foo.
19 %
20 % All definitions are read before anything is expanded, so you can use
21 % variables before they are defined.
22 %
23 % If a variable assignment is qualified with '.PROGRAM', it is ignored
24 % unless the current executable (last filename component of argv[0]) is
25 % named PROGRAM. This foo.PROGRAM construct is not recognized on the
26 % right-hand side. For environment variables, use FOO_PROGRAM.
27 %
```

```

28 % Which file formats use which paths for searches is described in the
29 % various programs' and the kpathsea documentation.
30 %
31 % // means to search subdirectories (recursively).
32 % A leading !! means to look only in the ls-R db, never on the disk.
33 % A leading/trailing/doubled ; in the paths will be expanded into the
34 % compile-time default. Probably not what you want.
35 %
36 % You can use brace notation, for example: /usr/local/{mytex:othertex}
37 % expands to /usr/local/mytex:/usr/local/othertex. Instead of the path
38 % separator you can use a comma: /usr/local/{mytex,othertex} also expands
39 % to /usr/local/mytex:/usr/local/othertex. However, the use of the comma
40 % instead of the path separator is deprecated.
41 %
42 % The text above assumes that path separator is a colon (:). Non-UNIX
43 % systems use different path separators, like the semicolon (;).
44
45 % Part 1: Search paths and directories.
46
47 % You can set an environment variable to override TEXMF if you're testing
48 % a new TeX tree, without changing anything else.
49 %
50 % You may wish to use one of the $SELFAUTO... variables here so TeX will
51 % find where to look dynamically. See the manual and the definition
52 % below of TEXMFCNF.
53
54 % The main tree, which must be mentioned in $TEXMF, below:
55 TEXMFMAIN = $SELFAUTOPARENT/texmf
56 % A place for local additions to a "standard" texmf tree.
57 TEXMFLocal = $SELFAUTOPARENT/texmf-local
58
59 % User texmf trees can be catered for like this...
60 HOMETEXMF=$HOME/texmf
61
62 % A place where texconfig stores modifications (instead of the TEXMFMAIN
63 % tree). texconfig relies on the name, so don't change it.
64 VARTEXMF = $SELFAUTOPARENT/texmf-var
65
66 % Now, list all the texmf trees. If you have multiple trees,
67 % use shell brace notation, like this:
68 % TEXMF = {$HOMETEXMF,!!$VARTEXMF,!!$TEXMFLocal,!!$TEXMFMAIN}
69 % The braces are necessary.
70 %
71 % A place where to store other TeX support files. It can be a remote
72 % texmf tree, or a tree to store non-free stuff, or ...
73 % TEXMFEExtra=$SELFAUTOPARENT/texmf-extra
74 % If you set this, add $TEXMFEExtra in the list below
75 %
76 TEXMF = {$HOMETEXMF,$TEXMFLocal,!!$VARTEXMF,!!$TEXMFMAIN}
77
78 % The system trees. These are the trees that are shared by all the users.
79 SYSTEXMF = $TEXMF
80
81 % The temporary area
82 TEMP = /var/tmp
83
84 % Where generated fonts may be written. This tree is used when the sources
85 % were found in a system tree and either that tree wasn't writable, or the
86 % varfonts feature was enabled in MT_FEATURES in mktex.cnf.
87 VARTEXFONTS = $VARTEXMF/fonts
88

```

```

89 % Where to look for ls-R files.  There need not be an ls-R in the
90 % directories in this path, but if there is one, Kpathsea will use it.
91 TEXMFDBS = $TEXMF
92
93 % It may be convenient to define TEXMF like this:
94 %   TEXMF = {$HOMETEXMF,!!$TEXMFLOCAL,!!$TEXMFMAIN,$HOME}
95 % which allows users to set up entire texmf trees, and tells TeX to
96 % look in places like ~/tex and ~/bibtex.  If you do this, define TEXMFDBS
97 % like this:
98 %   TEXMFDBS = $HOMETEXMF;$TEXMFLOCAL;$TEXMFMAIN;$VARTEXFONTS
99 % or mktexlsr will generate an ls-R file for $HOME when called, which is
100 % rarely desirable.  If you do this you'll want to define SYSTEXMF like
101 % this:
102 %   SYSTEXMF = $TEXMFLOCAL;$TEXMFMAIN
103 % so that fonts from a user's tree won't escape into the global trees.
104 %
105 % On some systems, there will be a system tree which contains all the font
106 % files that may be created as well as the formats.  For example
107 %   VARTEXMF = /var/lib/texmf
108 % is used on many Linux systems.  In this case, set VARTEXFONTS like this
109 %   VARTEXFONTS = $VARTEXMF/fonts
110 % and do not mention it in TEXMFDBS (but _do_ mention VARTEXMF).
111
112
113 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
114 % Usually you will not need to edit any of the other variables in part 1. %
115 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
116
117 % WEB2C is for Web2C specific files.  The current directory may not be
118 % a good place to look for them.
119 WEB2C = $TEXMF/web2c
120
121 % TEXINPUTS is for TeX input files -- i.e., anything to be found by \input
122 % or \openin, including .sty, .eps, etc.
123
124 % LaTeX-specific macros are stored in latex.
125 TEXINPUTS.latex = .;$TEXMF/tex/{latex,generic,}//
126 TEXINPUTS.hugelatex = .;$TEXMF/tex/{latex,generic,}//
127
128 % Fontinst needs to read afm files.
129 TEXINPUTS.fontinst = .;$TEXMF/{tex{/fontinst,},fonts/afm}//
130
131 % Plain TeX.  Have the command tex check all directories as a last
132 % resort, we may have plain-compatible stuff anywhere.
133 TEXINPUTS.tex = .;$TEXMF/tex/{plain,generic,}//
134 % other plain-based formats
135 TEXINPUTS.amstex = .;$TEXMF/tex/{amstex,plain,generic,}//
136 TEXINPUTS.ftex = .;$TEXMF/tex/{formate,plain,generic,}//
137 TEXINPUTS.texinfo = .;$TEXMF/tex/{texinfo,plain,generic,}//
138 TEXINPUTS.eplain = .;$TEXMF/tex/{eplain,plain,generic,}//
139
140 % e-TeX.
141 TEXINPUTS.elatex = .;$TEXMF/{etex,tex}/{latex,generic,}//
142 TEXINPUTS.etex = .;$TEXMF/{etex,tex}/{plain,generic,}//
143
144 % PDFTeX.  This form of the input paths is borrowed from teTeX.  A certain
145 % variant of TDS is assumed here, unaffected by the build variables.
146 TEXINPUTS.pdftexinfo = .;$TEXMF/{pdfetex,tex}/{texinfo,plain,generic,}//
147 TEXINPUTS.pdflatex = .;$TEXMF/{pdfetex,tex}/{latex,generic,}//
148 TEXINPUTS.pdfetex = .;$TEXMF/{pdfetex,tex}/{plain,generic,}//
149 TEXINPUTS.pdfelatex = .;$TEXMF/{pdfetex,pdfetex,etex,tex}/{latex,generic,}//

```

```

150 TEXINPUTS.pdfetex = .;$TEXMF/{pdfetex,pdftex,etex,tex}/{plain,generic,}//
151
152 % Omega.
153 TEXINPUTS.lambdax = .;$TEXMF/{omega,tex}/{lambda,latex,generic,}//
154 TEXINPUTS.omegax = .;$TEXMF/{omega,tex}/{plain,generic,}//
155
156 % Context macros by Hans Hagen:
157 TEXINPUTS.context = .;$TEXMF/{pdfetex,pdftex,etex,tex}/{context,plain,generic,}//
158
159 % cstex, from Petr Olsak
160 TEXINPUTS.cslatex = .;$TEXMF/tex/{cslatex,csplain,latex,generic,}//
161 TEXINPUTS.csplain = .;$TEXMF/tex/{csplain,plain,generic,}//
162 TEXINPUTS.pdfcslatex = .;$TEXMF/{pdfetex,tex}/{cslatex,csplain,latex,generic,}//
163 TEXINPUTS.pdfcsplain = .;$TEXMF/{pdfetex,tex}/{csplain,plain,generic,}//
164
165 % Polish
166 TEXINPUTS.platex = .;$TEXMF/tex/{platex,latex,generic,}//
167 TEXINPUTS.pdfmex = .;$TEXMF/{pdftex,tex}/{mex,plain,generic,}//
168 TEXINPUTS.mex = .;$TEXMF/tex/{mex,plain,generic,}//
169 TEXINPUTS.pdfplatex = .;$TEXMF/{pdftex,tex}/{platex,latex,generic,}//
170
171 % french
172 TEXINPUTS.frtex = .;$TEXMF/{mltex,tex}/{plain,generic,}//
173 TEXINPUTS.frlatex = .;$TEXMF/{mltex,tex}/{frlatex,latex,generic,}//
174
175 % MLTeX
176 TEXINPUTS.mltex = .;$TEXMF/{mltex,tex}/{plain,generic,}//
177 TEXINPUTS.mllatex = .;$TEXMF/{mltex,tex}/{latex,generic,}//
178
179 % odd formats needing their own paths
180 TEXINPUTS.lollipop = .;$TEXMF/tex/{lollipop,generic,plain,}//
181 TEXINPUTS.lamstex = .;$TEXMF/tex/{lamstex,generic,plain,}//
182
183 % David Carlisle's xmltex
184 TEXINPUTS.xmltex = .;$TEXMF/tex/{xmltex,latex,generic,}//
185 TEXINPUTS.pdfxmltex = .;$TEXMF/{pdfetex,tex}/{xmltex,latex,generic,}//
186
187 % Sebastian Rahtz' jadetex for DSSSL
188 TEXINPUTS.pdfjadetex = .;$TEXMF/{pdftex,tex}/{jadetex,generic,plain,}//
189 TEXINPUTS.jadetex = .;$TEXMF/tex/{jadetex,generic,plain,}//
190
191 % Earlier entries override later ones, so put this last.
192 TEXINPUTS = .;$TEXMF/tex/{generic,}//
193
194 % Metafont, MetaPost inputs.
195 MFINPUTS = .;$TEXMF/metafont/;{$TEXMF/fonts,$VARTHEXONTS}/source//
196 MPINPUTS = .;$TEXMF/metapost//
197
198 % Dump files (fmt/base/mem) for vir{tex,mf,mp} to read (see web2c/INSTALL),
199 % and string pools (.pool) for ini{tex,mf,mp}. It is silly that we have six
200 % paths and directories here (they all resolve to a single place by default),
201 % but historically ...
202 TEXFORMATS = .;$TEXMF/web2c
203 MFBASES = .;$TEXMF/web2c
204 MPMEMS = .;$TEXMF/web2c
205 TEXPOOL = .;$TEXMF/web2c
206 MFPOOL = .;$TEXMF/web2c
207 MPPOOL = .;$TEXMF/web2c
208
209 % Device-independent font metric files.
210 VFFONTS = .;$TEXMF/fonts/vf//

```

```

211 TFMFONTS = .;{$TEXMF/fonts,$VARTEXFONTS}/tfm//
212
213 % The $MAKETEX_MODE below means the drivers will not use a cx font when
214 % the mode is ricoh. If no mode is explicitly specified, kpse_prog_init
215 % sets MAKETEX_MODE to /, so all subdirectories are searched. See the manual.
216 % The modeless part guarantees that bitmaps for PostScript fonts are found.
217 PKFONTS = .;{$TEXMF/fonts,$VARTEXFONTS}/pk/{$MAKETEX_MODE,modeless}//
218
219 % Similarly for the GF format, which only remains in existence because
220 % Metafont outputs it (and MF isn't going to change).
221 GFFONTS = .;{$TEXMF/fonts/gf/$MAKETEX_MODE//
222
223 % A backup for PKFONTS and GFFONTS. Not used for anything.
224 GLYPHFONTS = .;{$TEXMF/fonts
225
226 % For texfonts.map and included map files used by mktexpk.
227 % See ftp://ftp.tug.org/tex/fontname.tar.gz.
228 TEXPONMAPS = .;{$TEXMF/fontname
229
230 % BibTeX bibliographies and style files.
231 BIBINPUTS = .;{$TEXMF/bibtex/{bib,}/}
232 BSTINPUTS = .;{$TEXMF/bibtex/{bst,}/}
233
234 % PostScript headers, prologues (.pro), encodings (.enc) and fonts;
235 % this is also where pdftex finds included figures files!
236
237 TEXPSEADERS.pdfplatex = .;{$TEXMF/{tex,pdftex,dvips,fonts/type1}//
238 TEXPSEADERS.pdfelatex = .;{$TEXMF/{tex,pdftex,dvips,fonts/type1}//
239 TEXPSEADERS.pdfetexinfo = .;{$TEXMF/{tex,pdftex,dvips,fonts/type1}//
240 TEXPSEADERS.pdfcsplatex = .;{$TEXMF/{tex,pdftex,dvips,fonts/type1}//
241 TEXPSEADERS.pdfcsplain = .;{$TEXMF/{tex,pdftex,dvips,fonts/type1}//
242 TEXPSEADERS.pdfetex = .;{$TEXMF/{tex,pdftex,dvips,fonts/type1}//
243 TEXPSEADERS.pdfjadetex = .;{$TEXMF/{tex,pdftex,dvips,fonts/type1}//
244 TEXPSEADERS.pdfplatex = .;{$TEXMF/{tex,pdftex,dvips,fonts/type1}//
245 TEXPSEADERS.pdfxmltex = .;{$TEXMF/{tex,pdftex,dvips,fonts/type1}//
246 TEXPSEADERS.pdfmex = .;{$TEXMF/{tex,pdftex,dvips,fonts/type1}//
247 TEXPSEADERS.pdfetex = .;{$TEXMF/{tex,pdftex,dvips,fonts/type1}//
248 TEXPSEADERS.pdfetexinfo = .;{$TEXMF/{tex,pdftex,dvips,fonts/type1}//
249 TEXPSEADERS.cont-de = .;{$TEXMF/{tex,pdftex,dvips,fonts/type1}//
250 TEXPSEADERS.cont-en = .;{$TEXMF/{tex,pdftex,dvips,fonts/type1}//
251 TEXPSEADERS.cont-nl = .;{$TEXMF/{tex,pdftex,dvips,fonts/type1}//
252 TEXPSEADERS.context = .;{$TEXMF/{etex,tex,pdftex,dvips,fonts/type1}//
253 TEXPSEADERS = .;{$TEXMF/{dvips,fonts/type1,pdftex}//
254
255 % PostScript Type 1 outline fonts.
256 T1FONTS = .;{$TEXMF/fonts/type1//;$TEXMF/fonts/misc/hbf//
257
258 % PostScript AFM metric files.
259 AFMFONTS = .;{$TEXMF/fonts/afm//
260
261 % TrueType outline fonts.
262 TTFONTS = .;{$TEXMF/fonts/truetype//
263 TTF2TFMINPUTS = .;{$TEXMF/ttf2pk//
264
265 % Type 42 outline fonts.
266 T42FONTS = .;{$TEXMF/fonts/type42//
267
268 % A place to puth everything that doesn't fit the other font categories.
269 MISCFONTS = .;{$TEXMF/fonts/misc//
270
271 % Dvips' config.* files (this name should not start with 'TEX!').

```

```

272 TEXCONFIG = .;$TEXMF/dvips//
273
274 % Makeindex style (.ist) files.
275 INDEXSTYLE = .;$TEXMF/makeindex//
276
277 % Used by DMP (ditroff-to-mpx), called by makempx -troff.
278 TRFONTS = /usr/lib/font/devpost
279 MPSUPPORT = .;$TEXMF/metapost/support
280
281 % For xdvi to find mime.types and .mailcap, if they do not exist in
282 % $HOME. These are single directories, not paths.
283 % (But the default mime.types, at least, may well suffice.)
284 MIMELIBDIR = $SELFAUTOPARENT/etc
285 MAILCAPLIBDIR = $SELFAUTOPARENT/etc
286
287 % TeX documentation and source files, for use with kpsewhich.
288 TEXDOCS = .;$TEXMF/doc//
289 TEXSOURCES = .;$TEXMF/source//
290
291 % allo for compressed files, and various extensions
292 TEXDOCSSUFFIX = .dvi:.ps:.html:.txt
293 TEXDOCSCOMPRESS = .:gz:.bz2:.zip:.Z
294 TEXDOCEXT = {$TEXDOCSSUFFIX}{$TEXDOCSCOMPRESS}
295
296 % Omega-related fonts and other files. The odd construction for OFMFonts
297 % makes it behave in the face of a definition of TFMFonts. Unfortunately
298 % no default substitution would take place for TFMFonts, so an explicit
299 % path is retained.
300 OFMFonts = .;$TEXMF/fonts,$VARTEXFonts}/{ofm,tfm};//;$TFMFonts
301 OPLFonts = .;$TEXMF/fonts,$VARTEXFonts}/opl//
302 OVFFonts = .;$TEXMF/fonts,$VARTEXFonts}/ovf//
303 OVPFonts = .;$TEXMF/fonts,$VARTEXFonts}/ovp//
304 OTPINPUTS = .;$TEXMF/omega/otp//
305 OCPINPUTS = .;$TEXMF/omega/ocp//
306
307 %% t4ht utility, sharing files with TeX4ht
308 TEX4HTFontset={alias,iso8859}
309 TEX4HTINPUTS = .;$TEXMF/tex4ht/base//;$TEXMF/tex4ht/ht-fonts/{$TEX4HTFontset}//
310 T4HTINPUTS= .;$TEXMF/tex4ht/base//
311 %% The mktex* scripts rely on KPSE_DOT. Do not set it in the environment.
312 KPSE_DOT = .
313
314 % This definition isn't used from this .cnf file itself (that would be
315 % paradoxical), but the compile-time default in paths.h is built from it.
316 % The SELFAUTO* variables are set automatically from the location of
317 % argv[0], in kpse_set_progname.
318 %
319 % About the /. construction:
320 % 1) if the variable is undefined, we'd otherwise have an empty path
321 %    element in the compile-time path. This is not meaningful.
322 % 2) if we used /$VARIABLE, we'd end up with // if VARIABLE is defined,
323 %    which would search the entire world.
324 %
325 % The TETEXDIR stuff isn't likely to be relevant unless you're using teTeX,
326 % but it doesn't hurt.
327 %
328 TEXMFCNF = .;{$SELFAUTOLOC,$SELFAUTODIR,$SELFAUTOPARENT}\
329 {,{$share,}/texmf{.local,}/web2c};c:/TeX/texmf/web2c
330
331
332 % Suggestions for editor settings under Windows. Uncomment your

```

```

333 % preferred option. The corresponding MFEDIT can also be set for use with
334 % Metafont.
335 %
336 % Winedt:
337 % TEXEDIT=C:\WinEdt\WinEdt.exe "[Open('%s');SelLine(%d,7)]
338 % Textpad:
339 % TEXEDIT = c:\Progra~1\TextPad\System\Ddeopn32 TextPad %s(%d)
340 % UltraEdit (newer Win32 versions):
341 % TEXEDIT = uedit32 %s/%d/1
342 % WinTeXShell32:
343 % TEXEDIT = texshell.exe /l=%d %s
344 % vi, vim, gvim. here we show Windows gvim.exe:
345 % TEXEDIT = gvim.exe %s +%d
346 % PFE:
347 % TEXEDIT=pfe32/g%d %s
348 % MED:
349 % TEXEDIT=med.exe "%s" %d
350 % TSE:
351 % TEXEDIT=e32.exe "%s" -n%d
352 % Epsilon (Lugaru) http://www.lugaru.com/
353 % TEXEDIT="c:\Program Files\eps90\bin\e32.exe" +%d %s
354
355 % For unix
356 %
357 % vi, vim, NEdit, (X)Emacs, pico, jed
358 % TEXEDIT = vi +%d %s
359 % TEXEDIT = vim +%d %s
360 % TEXEDIT = nedit +%d %s
361 % TEXEDIT = xemacs +%d %s
362
363 %(x)fte:
364 % TEXEDIT = xfte -l%d %s
365
366
367 %-----
368 % Write .log/.dvi/etc. files here, if the current directory is unwritable.
369 % TEXMFOUTPUT = /tmp
370
371 % If a dynamic file creation fails, log the command to this file, in
372 % either the current directory or TEXMFOUTPUT. Set to the
373 % empty string or 0 to avoid logging.
374 MISSFONT_LOG = missfont.log
375
376 % Set to a colon-separated list of words specifying warnings to suppress.
377 % To suppress everything, use TEX_HUSH = all; this is equivalent to
378 % TEX_HUSH = checksum:lostchar:readable:special
379 TEX_HUSH = none
380
381 % Enable system commands via \write18{...}?
382 shell_escape = f
383
384 % Allow TeX \openout/\openin on filenames starting with '.' (e.g., .rhosts)?
385 % a (any) : any file can be opened.
386 % r (restricted) : disallow opening "dotfiles".
387 % p (paranoid) : as 'r' and disallow going to parent directories, and
388 % : restrict absolute paths to be under $TEXMFOUTPUT.
389 openout_any = p
390 openin_any = a
391 % Allow TeX, MF, and MP to parse the first line of an input file for
392 % the %&format construct.
393 parse_first_line = t

```

```

394
395 % Enable the mktex... scripts by default? These must be set to 0 or 1.
396 % Particular programs can and do override these settings, for example
397 % dvips's -M option. Your first chance to specify whether the scripts
398 % are invoked by default is at configure time.
399 %
400 % These values are ignored if the script names are changed; e.g., if you
401 % set DVIPSMAKEPK to 'foo', what counts is the value of the environment
402 % variable/config value 'FOO', not the 'MKTEXPK' value.
403 %
404 % MKTEXTEX = 0
405 % MKTEXPK = 0
406 % MKTEXMF = 0
407 % MKTEXTFM = 0
408 % MKOCP = 0
409 % MKOFM = 0
410
411 % What MetaPost runs to make MPX files. This is passed an option -troff
412 % if MP is in troff mode. Set to '0' to disable this feature.
413 MPXCOMMAND = makempx
414
415
416 % Part 3: Array and other sizes for TeX (and Metafont and MetaPost).
417 %
418 % If you want to change some of these sizes only for a certain TeX
419 % variant, the usual dot notation works, e.g.,
420 % main_memory.hugetex = 20000000
421 %
422 % If a change here appears to be ignored, try redumping the format file.
423
424 % Memory. Must be less than 8,000,000 total.
425 %
426 % main_memory is relevant only to initex, extra_mem_* only to non-ini.
427 % Thus, have to redump the .fmt file after changing main_memory; to add
428 % to existing fmt files, increase extra_mem_*. (To get an idea of how
429 % much, try \tracingstats=2 in your TeX source file;
430 % web2c/tests/memtest.tex might also be interesting.)
431 %
432 % To increase space for boxes (as might be needed by, e.g., PiCTeX),
433 % increase extra_mem_bot.
434 %
435 % For some xy-pic samples, you may need as much as 700000 words of memory.
436 % For the vast majority of documents, 60000 or less will do.
437 %
438 main_memory = 263000 % words of inmemory available; also applies to inimf&mp
439 extra_mem_top = 0 % extra high memory for chars, tokens, etc.
440 extra_mem_bot = 0 % extra low memory for boxes, glue, breakpoints, etc.
441
442 % Words of font info for TeX (total size of all TFM files, approximately).
443 font_mem_size = 200000
444
445 % Total number of fonts. Must be >= 50 and <= 2000 (without tex.ch changes).
446 font_max = 1000
447
448 % Extra space for the hash table of control sequences (which allows 10K
449 % names as distributed).
450 hash_extra = 0
451
452 % Max number of characters in all strings, including all error messages,
453 % help texts, font names, file names, control sequences.
454 % These values apply to TeX and MP.

```

```

455 pool_size = 125000
456
457 % Minimum pool space after TeX/MP's own strings; must be at least
458 % 25000 less than pool_size, but doesn't need to be nearly that large.
459 string_vacancies = 25000
460 max_strings = 15000           % max number of strings
461 pool_free = 5000             % min pool space left after loading .fmt
462
463 % Hyphenation trie. As distributed, the maximum is 65535; this should
464 % work unless 'unsigned short' is not supported or is smaller than 16
465 % bits. This value should suffice for UK English, US English, French,
466 % and German (for example). To increase, you must change
467 % 'ssup_trie_opcode' and 'ssup_trie_size' in tex.ch (and rebuild TeX);
468 % the trie will then consume four bytes per entry, instead of two.
469 %
470 % US English, German, and Portuguese: 30000.
471 % German: 14000.
472 % US English: 10000.
473 %
474 trie_size = 64000
475
476 % Buffer size. TeX uses the buffer to contain input lines, but macro
477 % expansion works by writing material into the buffer and reparsing the
478 % line. As a consequence, certain constructs require the buffer to be
479 % very large. As distributed, the size is 50000; most documents can be
480 % handled within a tenth of this size.
481 buf_size = 200000
482
483 % These are Omega-specific.
484 ocp_buf_size = 20000         % character buffers for ocp filters.
485 ocp_stack_size = 10000      % stacks for ocp computations.
486 ocp_list_size = 1000       % control for multiple ocps.
487
488 % These work best if they are the same as the I/O buffer size, but it
489 % doesn't matter much. Must be a multiple of 8.
490 dvi_buf_size = 16384       % TeX
491 gf_buf_size = 16384       % MF
492
493 % It's probably inadvisable to change these. At any rate, we must have:
494 % 45 < error_line < 255;
495 % 30 < half_error_line < error_line - 15;
496 % 60 <= max_print_line;
497 % These apply to Metafont and MetaPost as well.
498 error_line = 79
499 half_error_line = 50
500 max_print_line = 79
501 stack_size = 300           % simultaneous input sources
502 save_size = 4000          % for saving values outside current group
503 param_size = 500          % simultaneous macro parameters
504 max_in_open = 15           % simultaneous input files and error insertions
505 hyph_size = 1000          % number of hyphenation exceptions, >610 and <32767
506 nest_size = 100           % simultaneous semantic levels (e.g., groups)
507 obj_tab_size = 200000     % PDF objects
508
509
510 main_memory.mpost = 1000000
511
512 main_memory.context = 1500000
513 hash_extra.context = 25000
514 pool_size.context = 750000
515 string_vacancies.context = 45000

```

```
516 max_strings.context = 55000
517 pool_free.context = 47500
518 nest_size.context = 500
519 param_size.context = 1500
520 save_size.context = 5000
521 stack_size.context = 1500
522 obj_tab_size.context = 256000
523
524 main_memory.hugetex = 1100000
525 param_size.hugetex = 1500
526 stack_size.hugetex = 1500
527 hash_extra.hugetex = 15000
528 string_vacancies.hugetex = 45000
529 pool_free.hugetex = 47500
530 nest_size.hugetex = 500
531 save_size.hugetex = 5000
532 pool_size.hugetex = 500000
533 max_strings.hugetex = 55000
534
535 main_memory.hugelatex = 1100000
536 param_size.hugelatex = 1500
537 stack_size.hugelatex = 1500
538 hash_extra.hugelatex = 15000
539 string_vacancies.hugelatex = 45000
540 pool_free.hugelatex = 47500
541 nest_size.hugelatex = 500
542 save_size.hugelatex = 5000
543 pool_size.hugelatex = 500000
544 max_strings.hugelatex = 55000
545 font_mem_size.hugelatex= 400000
546
547 main_memory.jadetex = 1500000
548 param_size.jadetex = 1500
549 stack_size.jadetex = 1500
550 hash_extra.jadetex = 50000
551 string_vacancies.jadetex = 45000
552 pool_free.jadetex = 47500
553 nest_size.jadetex = 500
554 save_size.jadetex = 5000
555 pool_size.jadetex = 500000
556 max_strings.jadetex = 55000
557
558 main_memory.pdfjadetex = 2500000
559 param_size.pdfjadetex = 1500
560 stack_size.pdfjadetex = 1500
561 hash_extra.pdfjadetex = 50000
562 string_vacancies.pdfjadetex = 45000
563 pool_free.pdfjadetex = 47500
564 nest_size.pdfjadetex = 500
565 save_size.pdfjadetex = 5000
566 pool_size.pdfjadetex = 500000
567 max_strings.pdfjadetex = 55000
568
569 main_memory.xmltex = 1500000
570 param_size.xmltex = 1500
571 stack_size.xmltex = 1500
572 hash_extra.xmltex = 50000
573 string_vacancies.xmltex = 45000
574 pool_free.xmltex = 47500
575 nest_size.xmltex = 500
576 save_size.xmltex = 5000
```

```
577 pool_size.xmltex = 500000
578 max_strings.xmltex = 55000
579
580 main_memory.pdfxmltex = 2500000
581 param_size.pdfxmltex = 1500
582 stack_size.pdfxmltex = 1500
583 hash_extra.pdfxmltex = 50000
584 string_vacancies.pdfxmltex = 45000
585 pool_free.pdfxmltex = 47500
586 nest_size.pdfxmltex = 500
587 save_size.pdfxmltex = 5000
588 pool_size.pdfxmltex = 500000
589 max_strings.pdfxmltex = 55000
590
591 font_mem_size.pdflatex = 210000
592 main_memory.pdflatex = 1500000
593 param_size.pdflatex = 3000
594 stack_size.pdflatex = 3000
595 hash_extra.pdflatex = 15000
596 string_vacancies.pdflatex = 45000
597 pool_free.pdflatex = 47500
598 nest_size.pdflatex = 500
599 pool_size.pdflatex = 500000
600 save_size.pdflatex = 5000
601 max_strings.pdflatex = 55000
602
603 main_memory.pdfelatex = 1500000
604 param_size.pdfelatex = 1500
605 stack_size.pdfelatex = 1500
606 hash_extra.pdfelatex = 15000
607 string_vacancies.pdfelatex = 45000
608 pool_free.pdfelatex = 47500
609 nest_size.pdfelatex = 500
610 pool_size.pdfelatex = 500000
611 save_size.pdfelatex = 5000
612 max_strings.pdfelatex = 55000
613
```

Summary: The T_EX Live Manual, 5th Edition

This article presents a translation of the T_EX Live manual into the Slovak language.

Sebastian Rahtz
sebastian.rahtz@oucs.ox.ac.uk
Michel Goossens
m.goossens@cern.ch

Seznámení s pojmy

Formát DVI

(Device independent) je soubor osmibitových čísel, která mohou být chápána jako instrukce a argumenty strojového jazyka. První byte každé instrukce je její operační kód a následující její argumenty¹. Aktuální font je reprezentován číslem typu *integer*, aktuální pozice v souboru je reprezentována kartézskými souřadnicemi (h,v). Souřadnice (0,0) určují levý horní roh. Zvětšování h resp. v odpovídá posunu vpravo resp. dolů. Nahlédneme-li na DVI jako na soubor instrukcí, které provedeme sekvenčně, dostaneme program, který proběhne přibližně následovně. Celý program je odstartován instrukcí *pre* a následně je definován font číslo 0. Každá strana začíná instrukcí *bop* a končí instrukcí *eop*. Mezi nimi jsou realizovány jednotlivé instrukce pro tisk jednotlivých znaků, změnu fontu a posuny v textu. Za poslední instrukcí *eop* následuje instrukce *post*, kterou je program ukončen.

Braillovo bodové písmo

Je bodové písmo, nejčastěji se sestáváví ze 2 sloupců obsahujících 3 řady bodů. Tedy 6 bodů na znak, což znamená $2^6 = 64$ znaků². Např. následující znak odpovídá písmenu 'a':

• ○
○ ○
○ ○

Pro malý počet možných písmen (max. 64) je zavedeno 6 prefixů, které mění význam znaků. Následující 2 znaky odpovídají písmenu 'A'³:

○ ○ ● ○
○ ○ ○ ○
○ ● ○ ○

- Definovány jsou prefixy pro změnu na malé nebo velké písmeno řecké abecedy, malé nebo velké písmeno latinské abecedy, na řetězec velkých písmen latinské abecedy a na číselný znak. Platnost prefixů pro řetězce je ukončena prázdným znakem (mezerou), nebo jiným prefixem.

¹Žádný, jeden nebo více.

²Každý bod může být vystouplý nebo hladký, proto 2^6 .

³První z nich je prefix pro velké písmeno latinské abecedy.

Pár pravidel pro přepis černotisku do Brailu

Číslice v Braillově písmu se zapisují písmeny a až j, před nimiž je zapsán prefix pro číselný znak. Pro oddělení tisíců používáme tečku a pro oddělení celé a desetinné části čísla čárku. Platnost prefixu pro čísla je ukončena mezerou, jiným prefixem, libovolným znakem mimo čárku, tečku a znaky a až j.

Shodně s černotiskem se zapisují řadové číslovky, datum, čas a měna. Římské číslice se zapisují jako řetězec velkých písmen. (I,V,X,L,C,D,M).

Vyznačení různých druhů písem

Začátek každého písmá, které je jiné než „běžné“ (kurzíva, tučné...), je označen dvojicí speciálních znaků, konec je pak označen dvojicí obrácenou.

- Například kurzíva

– Začátek

○ ○ ○ ●
○ ● ○ ○
● ○ ○ ●

– konec

○ ● ○ ○
○ ○ ○ ●
○ ● ● ○

DVIBRAILLE

Pro převod do Brailu je nutné udělat následující čtyři kroky. Převést DVI na holý text, zarovnat na 30–32 znaků na řádek a slova, která přesahují, rozdělit. Označkovat text a převést do znakové sady, která je vhodná pro tisk na braillové tiskárně. Pro převod DVI na holý text lze použít perlový modul *DVIParse* od Jana Pazdziory. V modulu jsou definována volání obslužných funkcí pro jednotlivé instrukce DVI. Ty pak stačí už „jen“ naprogramovat. Pro rozdělování slov existuje také modul od stejného autora, jmenuje se *hyphenation*. Definuje funkci *hyphenate*, které je jako argument předáno slovo a vrací pole čísel, která říkají, na kterém znaku je slovo možné rozdělit. Označkovat text je problém nejtěžší, neboť se značkuje podle sémantiky textu. Značkování běžných holých textů nabízejí zcela automaticky braillovy tiskárny, ale např. v případě textu s vyšší matematikou je nezbytný zásah lidské ruky. Uvedme příklad. Máme v textu máme zlomek $\frac{3}{4}$. Po převodu na holý text dostaneme zlomek zřejmě na 3 řádcích. A zde je kámen úrazu. Nevíme⁴, zda se jedná o zlomek, nebo se v textu

⁴Tedy my ano, ale program ne.

vyskytla podtržená trojka a pod ní náhodou čtverka. Oba tyto případy se samozřejmě značkují odlišně. Abychom ale zahnali chmury, je zde jedna potěšující skutečnost, totiž, že převod do jiné znakové sady je bezproblémový. Stačí jen znát výchozí a finální. Provést to můžeme např. skriptem *ctocs*.

Něco o programu

Program je napsaný v *perl* s využitím výše zmíněných balíčků od Jana Pazdziory. Je ke stažení na adrese <ftp://ftp.cstug.cz/pub/tex/local/cstug/tesar/dvibraille>. Nejedná se sice o žádný zázrak, program neumí převádět tabulky ani matematiku, rychlost není také nijak ohromující, ale pro převod běžných hladkých textů by měl být použitelný. Pro začátek každého fontu je vytištěna značka Z⁵, pro konec pak značka ~Z. Jako poslední parametr se skriptu předá jméno souboru a výsledek je pak uložen jako *jmeno_souboru.br*. Pokud je program spuštěn bez parametrů, vypíše se *help*.

Závěr

Závěrem bych rád poznamenal, že převod z DVI do Brailly se mi nejeví jako ideální díky výše zmíněným omezením, nicméně pro převod hladkých textů v podstatě vyhovuje.

Použitá literatura a software

1. Zdrojové texty programu *dvi2tty* zvláště pak soubor *dvistuff.c* ftp://ftp.fi.muni.cz/pub/linux/debian/dists/slink/non-free/binary-i386/tex/dvi2tty_5.1-9.deb.
2. Wanda Gonzúrová – Základní pravidla pro přepis běžných textů do bodového písma. Vydala a vytiskla Knihovna a tiskárna pro nevidomé K. E. Macata, Krakovská 21, Praha 1, v roce 1996.
3. Program *dvitype* a dokumentace *dvitype*.web <http://ftp.cstug.cz/pub/local/dviware/dvitype.zip>.
4. Moduly *DVIParse* a *HypHENation* jsou ke stažení na kterémkoliv serveru CPAN v adresáři *authors/id/JANPAZ/*.

Summary: DVIBRAILLE

To convert a DVI file into braille, we have to do these four steps.

⁵Nějaký ASCII znak

1. Convert a DVI file to a plain text.
2. Lines can be maximaly 32 characters long, so we have to split them and words, which are too long, have to be split too.
3. Put Brailles' marks into the text. (Mark the text.)
4. Convert the text to a character set which is suitable for printing on a braille printer.

The first step can be done with help of the perl module *DVIParse* (available from any CPAN site in the `authors/id/JANPAZ/` directory). The callbacks are defined for each of the DVI instructions in the module. Thus the only thing which you have to do is to program them. For the next step the perl module from the same author, called *hyphenation* (available on the same site as the *DVIParse*), can be used. There is defined the function *hyphenate* in the module. It takes a word as its argument and returns an array of numbers saying where the word can be hyphenated. The third step seems to be the hardest. Firstly we have to know the semantics of the text to put brailles' marks correctly, for example a fraction $\frac{3}{4}$. In the plain text the fraction takes 3 lines. The program doesn't know if it is a fraction or if there is underlined number 3 in the text and by chance there is number 4 under it. Each of these possibilities is marked differently. The last step is easy. You can do it with help of the script *ctocs* (in the case you are converting a czech text).

In my opinion this way is suitable only for converting a plain text not for structured text or text with higher mathematics.

T_EX Versioning System aneb jak všechny zdrojové soubory uložit

DAVID ANTOŠ

Tohle se vám už někdy muselo stát: Používáte T_EX v práci. Doma máte jinou distribuci. Doma jste pracně napsali a krásně zalomili dokument používající několik souborů s makry. Přišli jste do práce, zdroják na disketě, spustili T_EX a po chvíli I can't find file 'blabla.tex'. Naštvaně přemýšlíte, proč v instalaci v práci zrovna tenhle soubor chybí.

Nebo něco jiného. Máte vysázenou knihu, jasný bestseller, který určitě budete za rok či dva potřebovat vysadit znovu. Rádi byste si uložili všechny zdrojové

texty, na kterých vzhled dokumentu závisí. Takže se ponoříte do logu a hledáte je ručně. Tak to vás upřímně lituji, protože nevíte, že nemusíte.

Pro oba případy může posloužit **TVS**, **T_EX** Versioning System. Tento program umí projít logy **T_EX**u, najít v nich jména souborů (což zas není tak jednoduché) a tyto soubory zkopírovat do zadaného adresáře. Může najít nejen soubory potřebné přímo k překladu hlavního dokumentu, ale i soubory, které jsou potřeba k vygenerování formátu.

Přítom umí vynechávat soubory odpovídající regulárnímu výrazu – například se zadanými příponami. Navíc dokáže zacházet se jmény souborů docela inteligentně, dokáže zachovat strukturu vašich vlastních zdrojových souborů za předpokladu, že jsou umístěny v adresářové struktuře v podstromu adresáře, ve kterém je hlavní dokument.

Zatím experimentálně zařazujeme i možnost projít log **dvipsu**, který tento program vytváří většinou s volbou **-d**. Takto je možno získat fonty odkazované z **.dvi** souboru. Bohužel se takto dostanou pouze bitmapy, přesněji řečeno soubory **.pk**, což není příliš užitečné. Nicméně zkušený uživatel může získaný výpis použít jako vodítko pro ruční posbírání zdrojových souborů pro generování fontů.

Činnost **TVS** je samozřejmě limitována tím, co lze najít v logu. Bohužel trochu ruční práce zbývá, pokud dokument závisí na něčem, co lze sice považovat za zdrojový text, ale nepoužívá se to přímo v posledním běhu **T_EX**u. Příkladem může být soubor s bibliografickými údaji. Z něj se vyberou ty, na které je odkazováno z dokumentu a ty se třídí a doplní informacemi o stylu. Teprve tento derivát se začleňuje do dokumentu. Pokud chceme archivovat i původní soubor, nic nám nebrání ho k posbíraným zdrojům přidat.

Získané kompletní zdroje dokumentu je možno přeložit na jakékoli distribuci **T_EX**u. Stačí, aby cílová distribuce umožnila nastavit, aby **T_EX** hledal soubory nejdříve v aktuálním adresáři a jeho podadresářích a pak teprve v systémových adresářích **T_EX**u. Není třeba, aby systém **TVS** byl na cílovém stroji instalován.

TVS je napsán v Perlu ve snaze o maximální platformovou nezávislost. Je nezávislý na tom, jestli používáte **L^AT_EX**, plain nebo cokoli jiného. Zvládá konvence jmen souborů obvyklá v **M\$** světě i v **Un*xu**¹, počítá i s tak patologickými jevy, jako jsou mezery v názvech souborů, které se kupodivu vyskytují v **T_EX**u na Macu.

TVS jsem napsal jako bakalářský projekt na Fakultě informatiky Masarykovy university v Brně pod vedením a za vydatné podpory RNDr. Petra Sojky.

Domovská stránka je <http://www.fi.muni.cz/~xantos/TVS> a tamtéž si můžete **TVS** stáhnout včetně jeho podrobné dokumentace ve formátu Texinfo, na stránce je dokumentace dostupná také ve formátech HTML a PostScript. **TVS** je šířen zdarma za podmínek GPL licence.

¹měl jsem nutkání napsat „i v operačních systémech“ ...

Summary: T_EX Versioning System or How to Save All Source Files

TVS (T_EX Versioning System) collects all files needed to re-typeset T_EX documents exactly the same way on another T_EX installation. It is a platform-independent Perl script. TVS parses T_EX log files. This article discusses basic features of TVS, complete documentation is written in English.

Spolupráce databáze s L^AT_EXem

ZDENĚK WAGNER

Tento článek demonstruje výhody spojení databázového programu s L^AT_EXem. Předvádí, jak lze pomocí makrojazyka z téhož souboru generovat zcela odlišné výstupy s jiným využitím. Celý systém je předveden na programovém projektu, který je již řadu let využíván při organizaci konferencí CHISA.

Historie

Začátek použití výpočetní techniky při organizaci konferencí CHISA spadá již do šedesátých let. Tehdy to však byly počítače sálové, jejichž jediným vstupním médiem byly děrné štítky. Zpracování databáze, uložené na děrných štítcích, bylo tedy zcela neinteraktivní. Později byly údaje ukládány na magnetická média, ale vstupem nových dat i příkazů byly stále děrné štítky. Až koncem osmdesátých let se začaly používat osobní počítače, nejprve osmibitové s operačním systémem CP/M a později počítače šestnáctibitové se systémem MS-DOS. Konference měla tradičně okolo 1 500 příspěvků v několika odborných sekcích a v adresáři potenciálních zájemců bylo přibližně 16 000 záznamů. Počítač byl ovšem vybaven pouze jednotkami pro osmipalcové diskety, jejichž kapacita byla 256 KB. Taková databáze se na jednu disketu nevešla a komerční programy, které se navíc v tehdejších dobách sháněly velmi nesnadno, nedokázaly pracovat s databází rozložené na větším počtu disket. Byl proto vyvinut speciální databázový systém inspirovaný příkladem, který byl součástí kompilátoru Turbo-Pascal. Později byl tento systém přepsán do C++ a přeložen pro MS-DOS. Přes koncepční zastaralost se po mnoha postupných úpravách používá dodnes.

Správa takto rozsáhlé databáze vyžaduje značné úsilí a není pohodlné provádět všechny kontroly na obrazovce. Kontrolní výtisky však spotřebují velké

množství papíru. Je tedy nutné mít k dispozici větší množství formátů výstupních sestav, které obsahují pouze požadované údaje. Přidávání nových výstupních formátů však vyžadovalo změny v programu, který se tím stával stále nepřehlednějším se spoustou podmínek a přepínačů. Požadavky klienta nebylo možno uspokojovat na počkání a v některých případech si vyžádaly dokonce několika-denní ladění.

Po nástupu 24jehličkových tiskáren se organizátoři rozhodli, že je budou používat k tisku personalizovaných dopisů obsahujících údaje vytažené z databáze. Tehdy byl poprvé výstup z této databáze zpracován \LaTeX em, i když tisk více než tisíce dopisů trval několik dnů. Časový problém byl záhy vyřešen nástupem laserových tiskáren a \LaTeX ukázal své výhody při generování dalších typů výstupních sestav.

Po tomto historickém úvodu zapomeneme na databázové systémy, jejichž popis by patřil do zcela jiného časopisu. Zaměříme se na způsob, jak výstup z databáze dále zpracovat \LaTeX em. Vše, co bude dále řečeno, platí v zásadě i pro současné databázové systémy založené např. na dotazovacím jazyku SQL.

Požadované typy výstupů

Při správě databáze, jejíž záznamy jsou staré desítky let, nutně dochází k tomu, že některé z osob zemřou, jiné se přestěhují a mnozí změní zaměstnání a ztratí o následující konferenci zájem. Jindy zase někdo napíše svoji adresu trochu odlišným způsobem a dostane se tedy do databáze podruhé. Vzhledem ke stále rostoucím cenám poštovního a zvyšujícím se cenám tisku informačních letáků je nutno počet nežádoucích záznamů minimalizovat. To lze jen výjimečně automatizovat. Většinu práce musí provést člověk na základě vhodně strukturovaných informací vytištěných na papíře.

Nové údaje se dostávají do databáze z přihlášek na konferenci. Přestože řada z nich již pochází z interaktivních formulářů na WWW, stále vyžadují lidskou kontrolu. Protože požadavky na takovou kontrolu jsou jiné, musí výstupní formát být k tomu uzpůsoben.

Osobní počítač se postupně stává běžným vybavením domácností, data lze snadno přenášet na disketách i posílat prostřednictvím internetu. Přesto se stále s některými údaji musí pracovat i v místech, kde počítač není nebo dokonce být nemůže. Proto jsou nutné další typy výstupních sestav.

Programování takových sestav s využitím \LaTeX u je docela snadné. Nejprve musíme vytvořit výstupní soubor z databáze. Přitom využijeme prostředky databázového systému k tomu, aby záznamy setřídil v požadovaném pořadí. Databázové systémy to většinou zvládají lépe než jiné programy. Výstup naformátujeme tak, aby se pohodlně zpracovával \LaTeX em. (Lze samozřejmě použít i plain \TeX .) V uvedeném projektu je využit následující formát:

```

\def\cislo{0001}
\def\jmeno{Gorin}
\def\inic{A. V.}
\def\tit{Prof.}
\def\zeme{Russia}
\def\zemecesky{RU}
\def\zemenvel{RUSKO}
\def\tel{}
\def\fax{}
\def\email{}
\def\adri{...}
\def\adrii{...}
\def\adriii{...}
\def\adriv{}
\def\adrv{}
...
\exec

```

Jeden z typů výpisů pak zařídíme následujícím souborem (předchozí text je v souboru se jménem `dopis.tex`):

```

1 \NeedsTeXFormat{LaTeX2e}
2 \listfiles
3 \documentclass{article}
4 \usepackage{chivypis}
5
6 \begin{document}%\small
7 \AddressInput{dopis}
8 \end{document}

```

Všimněte si řádku 7. Záznamy v databázi totiž mohou obsahovat speciální znaky, které by mohly být $\text{T}_{\text{E}}\text{X}$ em interpretovány. Typickým znakem je `&`, který se vyskytuje zvláště v názvech firem. Soubor `chivypis.sty` vypadá takto:

```

9 \NeedsTeXFormat{LaTeX2e}
10 \ProvidesPackage{chivypis}[1998/03/18 Vypis for CHISA '98]
11 \pagestyle{empty}
12 \oddsidemargin -15mm
13 \topmargin -15mm
14 \textwidth 210mm
15 \advance\textwidth -2cm
16 \headheight 0pt
17 \headsep 0pt
18 \footskip 0pt
19 \textheight 297mm

```

```

20 \advance\textheight -2cm
21 \tolerance 9999
22 \hbadness \tolerance
23 \clubpenalty 10000
24 \widowpenalty 10000
25 \interlinepenalty 8888
26 \raggedright
27
28 \def\AddressInput#1{\begingroup
29 \@sanitize\catcode'\=0 \catcode'\ =10 \relax
30 \input #1
31 \endgroup}
32
33 \newif\ifviceautoru
34 \newif\ifvicetypu
35 \newif\ifLP
36
37 \def\initadr{\@bsphack
38   \let\inic\empty
39   \let\jmeno\empty
40   \let\tit\empty
41   \let\adri\empty
42   \let\adrii\empty
43   \let\adriii\empty
44   \let\adriv\empty
45   \let\adrv\empty
46   \let\zeme\empty
47   \let\zemecesky\empty
48   \let\zemenvel\empty
49   \let\cislo\empty
50   \let\autori\empty
51   \let\nazev\empty
52   \let\topic\empty
53   \let\typ\empty
54   \let\afili\empty
55   \let\afiliii\empty
56   \let\afiliiii\empty
57   \let\afiliv\empty
58   \global\viceautorufalse
59   \global\vicetypufalse
60 \@esphack}
61 \initadr

```

```

62
63 \def\afiliace{\afili
64 \ifx\afilii\empty\else; (1)\afilii\fi
65 \ifx\afiliii\empty\else; (2)\afiliii\fi
66 \ifx\afiliv\empty\else; (3)\afiliv\fi}
67
68 \def\ViceAutoru{\global\viceautorutruue}
69 \def\ViceTypu{\global\vicetyputruue}
70
71 \def\typ@L{lecture}
72 \def\typ@P{poster}
73 \def\typ@LP{\LPtrue lecture or poster}
74 \def\typ@ES{MAPRET lecture}
75 \def\typ@M{MAPRET lecture}
76 \def\typ@K{keynote lecture}
77 \def\typ@X{plenary lecture}
78
79 \def\poster{P}
80 \def\plenary{X}
81 \def\lecture{L}
82
83 \def\theshemtyp{\csname typ@\shemtyp\endcsname}
84
85 \def\delim{\unskip~$|\$ \ignorespaces}
86
87 \def\exec{\LPfalse\ifx\typ\empty \def\typ{LP}\fi
88 \ifnum\shemdur>0
89 \begingroup
90 \noindent \hangafter 1 \hangindent 1em \textbf{\cislo:}
91 \autori \delim \afiliace \delim
92 \textbf{\csname shemid\endcsname} \delim \nazev \par
93 \endgroup \initadr \fi}

```

Makro `\AddressInput`, definované na řádce 28, nastaví kategorii většiny znaků na 12. Některým znakům však musíme vrátit kategorii původní. V makru je tím ošetřena většina případů, ale ne všechny. Typickým překlepem je { místo P (je to sousední klávesa), což způsobí fatální chybu při tisku. Jedinou možností nápravy je spolupráce s databázovým systémem, který nesmí povolit složené závorky v žádném poli.

Nemáme jistotu, zda výstup z databáze bude inicializovat i prázdné položky. Pro jistotu si počínáme řádkem 37 nadefinujeme příslušné makro, které použijeme po každém záznamu. Ve stručném výpisu budeme záznamy oddělovat matema-

tickým znakem definovaným na řádku 85. Výpis každého záznamu je proveden makrem `\exec`, které je definováno na řádku 87. Všimněte si, že používáme podmínku `\ifnum` pro potlačení výpisu nežádoucích záznamů, které nebylo možno vyloučit pomocí prostředků použitého databázového systému.

Nebudeme zde popisovat význam všech pomocných maker. Ta jsou totiž silně závislá na použitém databázovém systému a na obsahu databáze. Navíc se požadavky mění. Uvedený příklad pochází z roku 1998, v letech předchozích i následujících byla některá makra trochu jiná.

Změnou definice maker lze snadno docílit jiného typu výpisu. Ostatně, jak napovídá název souboru, původně byl tento přístup využit pro tisk dopisů. Pro tento účel jsou pak makra mírně modifikována:

```

94  \def\recatcode{%
95    \catcode'\$=3
96    \catcode'\&=4
97    \catcode'\#=6
98    \catcode'\^=7
99    \catcode'\_ =8
100   \catcode'\~=13
101   \catcode'\%=14}
102
103  \newcount\c@test@page
104  \def\setZWpage{\ifhmode\par\fi \global\c@test@page=\c@page}
105  \def\testZWpage{\ifhmode \par \fi
106    \ifnum \c@test@page=\c@page
107      \else \typeout{*** Letter \cislo\space too long ***}\fi}
108
109  \def\exec{\LPfalse\ifx\typ\empty \def\typ{LP}\fi
110    \ifnum\shemdur>0
111    \begingroup
112      \setZWpage
113      \recatcode\tolerance 9999 \hbadness\tolerance
114      \brokenpenalty 10000
115      \lethead \dopis
116      \testZWpage \clearpage \endgroup \initadr \fi}

```

V některých případech byl při rozvoji makra `\exec` načítán další soubor, který obsahoval \TeX ové příkazy. Proto máme na řádku 94 definováno makro, jež obnoví potřebné kategorie. Dále potřebujeme zkontrolovat, že se dopis vždycky vejde na stránku. Na řádku 103 si proto deklaruujeme pomocný čítač a definujeme testovací makra. \LaTeX ovými prostředky generujeme hlavičkový papír. Příslušná hlavička je definována v makru `\lethead`. Vlastní text dopisu je v makru `\dopis`. Pro zvýšení modularity bylo toto makro vždy definováno v jiném sty-

lovém souboru. Každý z autorů pak dostal potvrzovací dopis, že přihláška došla a později dopis, že jeho příspěvek byl přijat. Do makra `\shendur` ukládá databázový systém délku přednášky v minutách, u vývěsek obsahuje libovolnou kladnou hodnotu. Nepřijaté příspěvky mají toto pole vynulované. Pak lze drobnou úpravou využívající podmínku `\ifnum` tisknout zamítací dopisy.

Na organizaci konference se podílí mnoho lidí a každý z nich je zodpovědný za určitou část odborného programu. Aby se jim pracovalo pohodlněji, vyžadují potřebné informace v elektronické podobě, aby si je mohli načíst do jiné databáze nebo jen do textového editoru. Využívají se k tomu následující dva postupy:

1. Informace jsou zpracovány \LaTeX em do vhodného formátu. Programem `dvispell`, jenž je standardní součástí $\text{em}\TeX$ u, je exportován prostý text.
2. Využije se pevného formátu záznamů. Víme, že záznam obsahuje (až na výjimky) pouze definice maker a končí makrem `\exec`. Takový soubor lze pak snadno zpracovat skriptovacím jazykem, jakým je Rexx nebo Perl. Přitom se pracuje pouze s jednoduchými řetězcovými operacemi, regulární výrazy nejsou nutné.

Příprava odborného programu konference

Abychom si mohli vysvětlit další část projektu, musíme se zmínit o tom, jaké úkoly jsou požadovány. Každá přihláška je posouzena programovou komisí, je rozhodnuto, zda příspěvek bude prezentován formou přednášky či vývěsky, a je mu přiděleno pevné místo v některé z odborných sekcí. Kromě původního chronologicky přidělovaného čísla, které se přenáší do makra `\cislo`, získá příspěvek další identifikaci. Údaje se zadávají v textovém souboru pomocí původních čísel, speciálních značek a předdefinovaných \LaTeX ových maker. Syntaxi nebudeme popisovat, uvedeme si jen malou ukázkou:

```
*X1L083Plenary lectures
```

```
:45
```

```
1269 1219
```

```
*X3L083
```

```
1276 1270
```

```
*X5L083
```

```
167 1268
```

```
*X7L083
```

1281 1266

```
*A1L093Reaction engineering
\subject{Reaction kinetics}
\chairman{M. Marek, P. L. Silveston}
```

K 1380/30
:30 51 359 410 505 1139 1167

```
*A2L140Reaction engineering
\subject{Chemical reactors\,---\,dynamics and modelling}
\chairman{J. Markoš, T. Salmi}
```

K 712/30
:30 1032 1049 284 431 60

Specializovaný program kontroluje, že každé číslo je uvedeno v tomto souboru právě jednou. Autorský rejstřík odkazuje pouze na čísla referátů, nikoliv na čísla stránek. Navíc je používán ke kontrolním účelům zcela samostatně. Jeho vytvoření je proto svěřeno dalšímu specializovanému programu, který získává informace z databáze a sám provádí třídění. Přitom algoritmus je poněkud upraven, protože cizinci by písmena s diakritikou zařazená podle českých zvyklostí pravděpodobně nenašli. Získáme tedy dva výstupy. První soubor obsahuje makra potřebná pro sazbu vlastního odborného programu. Vypadá následovně (vybíráám začátek sekce A, protože je zajímavější, a dlouhé řádky byly ručně zlomeny, aby se vešly do formátu Zpravodaje):

```
\newsect{LA1:Reaction engineering}
\subject{Reaction kinetics}

\chairman{M. Marek, P. L. Silveston}

\RefType{K}
\numtime{A1.1}{9}{30}
\authors{M. A. Panthaky, R. V. Vladea, R. R. Hudgins,
\main P. L. Silveston}
\afil{Univ. Waterloo, Ont., CA}
\nazev{Unique low-temperature sulfuric acid
process.} \orignum[1380]

\RefNormal
\numtime{A1.2}{10}{00}
\authors{\main M. I. Nunes, M. M. Dias, J. C. B. Lopes}
```

```
\afil{Univ. Porto, PT}
\nazev{Mixing assessment and accurate dead-time
determination of reaction kinetics with stopped-flow
spectrophotometry.} \orignum[51]
```

Zde je začátek souboru pro tisk autorského rejstříku:

```
\begin{autindex}
```

```
\aboveXskip
```

```
\hfill{\bfseries A}\hfill
```

```
\belowXskip\par
```

```
\item Aatmeeyata,\pfill \normalref D2.6[101]\par
\item Abaskuliev, J. A.\pfill \normalref A7.6[328]\par
\item Abbas, A. S.\pfill \normalref A7.5[805]\par
\item Abbasova, M. T.\pfill \normalref P5.164[786]\par
\item Abd-Rabbo, M. F.\pfill \normalref F1.2[141]\par
\item Abdel-Shafi, N. Y.\pfill \mainref E8.2[118],
\normalref E8.3[119]\par
\item Abfalterer, T.\pfill \mainref D3.3[708]\par
\item Abo El-Lail, A. S.\pfill \normalref E8.3[119],
\normalref G1.6[120], \mainref G1.7[121]\par
\item Aboltins, A.\pfill \normalref P3.111[469]\par
\item Ačai, P.\pfill \normalref A5.1[362],
\mainref P5.80[391]\par
\item Achouri, A.\pfill \normalref P5.169[610]\par
\item Acierno, D.\pfill \normalref P7.137[653]\par
\item Adler, P. M.\pfill \normalref A2.5[431]\par
\item Adnadjevic, B.\pfill \normalref P3.88[164],
\mainref P5.12[354], \mainref P3.87[355]\par
```

Vzorovou stránku odborného programu vidíte na obrázku 1, vzorová stránka autorského rejstříku je na obrázku 2. Původní čísla, která jsou ve zdrojovém souboru pro autorský rejstřík v hranatých závorkách, nejsou zdánlivě využita. Jejich význam je ovšem dvojnásobný. V určité fázi kontrol je nutné znát přiřazení obou typů identifikací. V kontrolním autorském rejstříku pro vnitřní potřebu je pak skutečně vytištěna informace typu „D2.6[101]“. Finální verze je vytvářena jak v tištěné podobě, tak ve formátu PDF. Elektronická verze je pak umístěna na CD společně s texty přednášek a vývěsek (ty jsou také v PDF). Jména PDF souborů s plnými texty jsou odvozena právě z původních čísel příspěvků. Jejich přítomnost ve vstupním souboru pro tisk autorského rejstříku tedy umožňuje vytvoření hypertextových odkazů na plné texty.

PLENARY LECTURES

Monday

A1.0 8.30 Future of chemical engineering: chemical, process and product-oriented engineering. **J.-C. Charpentier** (Ec. Sup. Chim. Phys. Elec. CNRS, Villeurbanne, FR) [1269]

H1.0 8.30 The role of universities in integrating the "two cultures": art and science. **A. Tamir** (Ben-Gurion Univ. Negev, Beer Sheva, IL) [1219]

Tuesday

A3.0 8.30 Chemical research in the EC programme Competitive and Sustainable Growth. **J.-C. Toussaint** (European Commission, Bruxelles, BE) [1276]

H3.0 8.30 Computational fluid dynamics: more than promise to chemical engineering. **H. Van den Akker** (Delft Univ. Technol., NL) [1270]

Wednesday

A5.0 8.30 The education of chemical engineers in the third millennium. **J. E. Gillett** (Macclesfield, UK) [167]

H5.0 8.30 Quo vadis the analysis of time series in chemical engineering? **J. Drahoš** (Inst. Chem. Proc. Fundam., Praha, CZ) [1268]

Thursday

A7.0 8.30 Sustainable industrial development: myth or reality? **D. Bricknell** (CEFIC, Bruxelles, BE) [1281]

H7.0 8.30 Experiences with product engineering. **J. A. Wessel- ingh** (Univ. Groningen, NL) [1266]

A

A1 Lectures—Monday morning Reaction engineering

Reaction kinetics

Chairmen: M. Marek, P. L. Silveston

A1.1 9.30 Keynote lecture: Unique low-temperature sulfuric acid process. **M. A. Panthaky, R. V. Viadea, R. R. Hudgins, P. L. Silveston** (Univ. Waterloo, Ont., CA) [1380]

A1.2 10.00 Mixing assessment and accurate dead-time determination of reaction kinetics with stopped-flow spectrophotometry. **M. I. Nunes, M. M. Dias, J. C. B. Lopes** (Univ. Porto, PT) [51]

A1.3 10.30 Improving the kinetic model for tert-amyl-methyl ether synthesis by considering the reaction medium. **L. K. Rihko-Struckmann, P. Latostenmaa, A. O. I. Krause** (Helsinki Univ. Technol., FI) [359]

A1.4 11.00 Comparison of coal pyrolysis kinetics estimated by thermogravimetric and fluidized bed experiments. **V. Barisic, S. Oka, M. Ilic** (Inst. Nucl. Sci. Vinca, Beograd, YU) [410]

A1.5 11.30 In-line conversion monitoring in pmma high conversion polymerization process. **T. Zeilmann, L. Cavin¹, T. Meyer** (Swiss Fed. Inst. Technol., Lausanne, CH); ¹Givaudan Roure, Vernier, CH) [505]

A1.6 12.00 Utilization of supported ruthenium catalyst for improved production technology on the three-phase hydrogenation of xylose to xylitol. **J.-P. Mikkola, T. Salmi, A. Kalantar, E. Toukonlitty** (Abo Akademi, Turku, FI) [1139]

A1.7 12.30 Wet oxidation of activated sewage sludge. **R. Zarzycki, M. Imbierowicz, A. Zawadzka** (Tech. Univ. Lodz, PL) [1167]

A2 Lectures—Monday afternoon Reaction engineering

Chemical reactors - dynamics and modelling

Chairmen: J. Markos, T. Salmi

A2.1 14.00 Keynote lecture: Development of rate expressions and parameter estimation strategies for catalytic two- and three-phase processes. **T. Salmi, J.-P. Mikkola, J. Wärna, M. Rönnholm** (Abo Akad., FI) [712]

A2.2 14.30 Kinetic modelling of single coal particle combustion. **L. Jelemenský, J. Markos, R. Zajdlík, B. Remiarová** (Slovak Univ. Technol., Bratislava, SK) [1032]

A2.3 15.00 Analysis of process nonlinearities for estimating the performance of periodically operated chemical reactors. **R. Gutsche, R. Lange¹, J. Frauhammer², W. Witt** (BTU Cottbus, DE); ¹M. L. Univ., Halle-Wittenberg, DE; ²Univ. Stuttgart, DE) [1049]

A2.4 15.30 Modeling and multi-objective optimization of an industrial wiped-film poly(ethylene terephthalate) reactor. **V. Bhaskar, S. K. Gupta, A. K. Ray** (Nat. Univ. Singapore, SG) [284]

A2.5 16.00 Mesoscale modeling of gas-liquid-solid catalytic processes. **F. Štěpánek, P. M. Adler¹, M. Marek** (Inst. Chem. Technol., Praha, CZ); ¹IPGP, Paris, FR) [431]

A2.6 16.30 A transient model for the performance of dual fuel (natural gas/diesel) catalytic converter. **N. Sallamie, M. Kazemini¹, A. Badakhshan², M. Soltanien³, M. Estiri³** (Iran Univ. Sci. Technol., Tehran, IR); ¹Sharif Univ. Technol., Tehran, IR; ²Univ. Calgary, CA; ³Air Quality Control Co., Tehran, IR) [60]

Obrázek 1: Vzor stránky odborného programu konference CHISA 2000

A

Aatmeeyata,	D2.6	Amaral, A. L.	P1.32
Abaskuliev, J. A.	A8.6	Amazouz, M.	P1.150
Abbas, A. S.	A8.5	Ambros, F.	P1.40
Abd-Rabbo, M. F.	F1.2	Ambrozek, B.	P3.105, P1.114
Abdel-Shafi, N. Y.	E8.2, E8.3	Ambrozy, E.	E4.6
Abdelbaki, N.	E6.5, P7.200	Amidpour, M.	H4.6, H4.2
Abfalterer, T.	D3.3	Amil, C.	P3.150
Abo El-Lail, A. S.	E8.3, G1.5, G1.6	Amireche, F.	P3.191
Aboltins, A.	P3.128	Amiri, M. C.	B8.7, B2.7
Ácái, P.	A5.1, P5.79	Anderson, B. G.	P3.131
Achouri, A.	P5.183	András, C.	P3.64
Acierno, D.	P7.173	Andreeva, D.	P3.148
Adler, P. M.	A2.5	Angelov, G.	P3.49, P3.37
Adnadjevic, B.	P5.12	Angelova, B.	J4.5, J4.2 , J4.1
Adonyi, R.	P7.71	Anikeev, V. I.	B5.8
Adrien, R. J.	I2.4	Anikeev, V.	B6.8
Adris, A.	H4.3	Anne-Archard, D.	G8.5
Afanasjev, A. N.	P7.14	Annus, J.	P5.131
Agachi, S. P.	P7.6, P7.7	Ansourian, P.	F1.1
Agachi, S.	P5.53, P3.12	Antonopoulos, D. K.	H7.1
Agacinski, P.	F1.67	Antonovich, D.	P7.153
Agafonov, G. I.	P7.217	Anutriv, L. P.	P5.176
Agamina, A. V.	P3.77	Aoun-Habbache, M.	F4.5
Agar, E.	F1.3	Apel, P. Yu.	P3.163 , D3.4
Agoudjil, N.	P3.169	Apostel, R.	D2.3
Aguado, R.	P5.157	Ar, I.	P3.92, B3.2
Aguilera, M. E.	P1.14	Arandes, J. M.	P5.12
Aim, K.	C1.4 , C3.5	Aranzabal, A.	I4.4, P7.156
Aiouache, F.	J1.3	Ardjmand, M.	A5.7, P5.14 , P3.31
Ait-Amar, H.	P5.51 , P7.137	Arest-Yakubovich, A. A.	P5.32
Akay, G. B7.5, I3.4, P7.110, P3.220, P3.150, P5.170, F5.1, D4.8		Arévalo, S.	P3.209
Akbarnejad, M. M.	P5.1 , B3.5, P5.24	Aris, M. J.	P7.163
Akhmadiev, F. G.	P1.77	Aristov, Yu. I.	P3.120, P3.121, P3.122, P3.123
Akhetmetov, T. G.	P7.203	Arlamova, N. T.	P3.47
Akiya, T.	E7.5	Aroua, M. K.	P3.29, P7.163, F1.4, J7.5
Akman, U.	G1.3, H7.2 , H7.3, H7.4	Artemov, A. V.	P5.101
Aknoun, A.	P3.98	Artemov, I. L.	G5.6
Akporiaje, D.	J2.3	Arutyunov, V. S.	A8.3, P5.9, P5.10
Aksakal, B.	P7.110, P3.167, P1.106	Asako, Y.	E5.4
Aksenova, E. L.	P3.156	Aso, K.	C8.2
Al Khaddour, A. M.	I6.8	Asy, M. A.	P1.167
Al-Ahmad, M. I.	E1.2	Atabek, B.	P5.165, P5.166
Al-Masry, W. A.	E1.2	Atakül, H.	E5.6
Al-Otaibi, M. A.	H4.3	Atala, D. I. P.	P5.57
Al-Riyami, B. A.	P7.60	Atamanenko, I. D.	P3.165
Al-Zubiedy, A. M.	P1.92	Atamanenko, I.	P3.172
Alaoui, J. B.	G3.4	Atik, Y.	P1.106
Alakent, B.	H6.7	Atik, Z.	P3.191
Albrecht, W.	D3.2, P3.160 , D1.1	Aurelle, Y.	P7.113, P7.114, P7.147 , P7.146
Alejski, K.	C7.5	Avetisov, A. K.	P5.116
Aleksic, R.	P1.100	Avramenko, Yu.	D2.5
Alemzadeh, I.	P7.174	Avramova, T.	J4.5
Alexandre, R. G.	P1.10	Avseev, A. V.	P7.14
Alexandrova, S.	P1.58, P3.23, P3.153, P1.59, P3.52	Axente, D.	B3.4
Algur, O. F.	P7.119	Ayyad, A.	G1.5 , G1.6
Allenov, D. N.	P1.163	Azapagic, A.	J3.8 , K5.1
Allgower, F.	P3.12	Aziz, A. A. A.	P7.155
Aloisio, L.	B5.7	Aznar, M.	P3.197
Altinov, S. V.	P1.104	Azouaou, N.	P3.48
Alvarez, J. I.	P7.156		
Alvarez, M. M.	E3.6, E2.4		
Alvarez, S.	P1.30		
Alves, M. M.	P7.167		
Alymova, A. T.	P7.126		

B

Babák, Z.	P5.140
Babić, A.	C1.4
Babic, M. D.	B8.2, B8.3
Babinec, F.	I5.1 , P7.89, P7.90, I5.3

Obrázek 2: Vzor stránky autorského rejstříku konference CHISA 2000

Makra pro sazbu autorského rejstříku zde nebudeme popisovat. Jsou inspirována stylem DOC.STY (nyní je to balík LTXDOC). V tomto balíku lze nalézt velmi dobrý zdroj inspirace pro sazbu libovolných rejstříků a rozhodně jej doporučuji ke studiu.

Makra pro sazbu odborného programu jsou dosti složitá. Ukážeme si jen vybranou část. Začneme makrem, které se vlastně volá až na konec. Předběžný program je připravován pouze v tištěné verzi a v PDF je poskytován pouze na Internetu. Texty přednášek v té době nejsou ještě k dispozici a hypertextové odkazy tudíž nemají smysl. Navíc nejsou ještě pevně přiřazené identifikace, jedinou rozlišovací značkou jsou původní chronologicky přidělovaná čísla. Přesněji řečeno, databázový program nějaká označení přidělí, ale při sazbě předběžného programu se ignorují. Makro `\LinkToFile` (řádek 119) tedy pouze vytiskne svůj argument v hranatých závorkách a makro `\OtherLinkToFile` (řádek 120) svůj argument spolkne. Ve finálním programu však jsou již identifikace, obsahující označení sekce, velmi důležité. Makro `\final`, jehož definice začíná na řádce 129, předefinuje určitá formátovací makra. Hypertextové odkazy na PDF soubory plných textů jsou generovány s využitím maker z balíku PDFHDR.STY, který byl popsán v [1]. Použité makro `\four@digits` pouze doplňuje vedoucí nuly tak, aby číslo bylo čtyřmístné. Jeho definici vidíte na řádce 122. Makro `\chisa@file` na řádce 127 konvertuje čtyřmístné číslo na jméno souboru. Autoři CD si nejsou jisti, zda norma ISO 9660 neomezuje maximální počet souborů v adresáři na CD. Proto je v každém adresáři pouze 100 souborů, přičemž jméno adresáře je tvořeno prvními dvěma číslicemi. Všimněte si, že definice maker, které jsou obsaženy uvnitř makra `\final`, používají řadu primitivů `\expandafter`. Ty zajišťují, že se vše expanduje ve správném pořadí.

```
117 \let\StartRef\relax
118
119 \def\LinkToFile#1{[#1]}
120 \def\OtherLinkToFile{}
121
122 \def\four@digits#1{%
123   \ifnum #1<1000 0\fi
124   \ifnum #1<100 0\fi
125   \ifnum #1<10 0\fi \the#1}
126
127 \def\chisa@file#1#2#3#4{#1#2/#1#2#3#4.pdf}
128
129 \def\final{%
130   \def\LinkToFile##1{\count@=##1
131     \edef\zw@fname{\four@digits\count@}%
132     \expandafter\pdffile\expandafter{\expandafter
```

```

133     \chisa@file\zw@fname}{[#1]}}
134 \def\OtherLinkToFile #1[#2]{\count@##2
135     \edef\zw@fname{\four@digits\count@}%
136     \expandafter\pdf\file\expandafter{\expandafter
137     \chisa@file\zw@fname}{#1}}
138 \let\plenary@days\zw@days
139 \setcounter{tocdepth}{1}%
140 \def\room##1 {}%
141 \def\numtime##1##2##3{\def\StartRef{\startit
142     \hbox to 5em{\bfseries##1\hss ##2.##3}%
143     \hbox to .5em{\hss}}}%
144     \def\num##1{\def\StartRef{\startit \getnumsize{##1}%
145     \hbox to \num@dim{\bfseries##1\hss}\enspace
146     \ignorespaces}}%
147 \def\CoffeeBreak##1##2{\startit
148     {\bfseries##1.##2}\enspace Coffee break}}

```

Následující makra definují vzhled jednotlivých záznamů. Všimněte si, že jednotlivé položky jsou tištěny v jiném pořadí, než v jakém se vyskytují ve zdrojovém souboru. Proto si je musíme nejprve uschovat v pomocných makrech a skutečný tisk provádí až `\orignum`. Makro `\sz` použité na řádce 154 pouze přepíná velikost písma, `\zw@cat` mění kategorie mnoha znaků podobně jako již dříve popísované makro `\AddressInput` (řádek 7).

```

149 \newcommand\startit{\ifhmode \par \fi
150     \vskip \baselineskip \noindent \ignorespaces}
151
152 \newcommand\main{\raisebox{.07ex}[z@][z@]{*}%
153     \ignorespaces }
154 \newcommand\af[1]{\raisebox{.8ex}[z@][z@]{\sz #1}}
155 \newcommand\authors{\begingroup
156     \zw@cat \catcode'\.\active \zw@aut}
157 \newcommand\zw@aut[1]{\gdef\Aut{#1}\endgroup
158     \ignorespaces }
159 \newcommand\afil{\begingroup \zw@cat
160     \catcode '\.\active \zw@afil}
161 \newcommand\zw@afil[1]{\gdef\Afil{(#1)}\endgroup
162     \ignorespaces}
163 \newcommand\nazev{\begingroup \zw@cat \zw@nazev}
164 \newcommand\zw@nazev[1]{\gdef\Naz{#1}\endgroup
165     \ignorespaces}
166 \newcommand\orignum[1][1]{\the@lbl{#1}%
167     \let\the@lbl@\gobble \csname at #1\endcsname \StartRef

```

```

168   \theRefType \Naz\ {\bfseries\Aut} \Afil
169   \unskip~\LinkToFile{#1}\par \relax
170   \csmame after #1\endcsname\relax}
171
172   \let\the@lbl\@gobble

```

Písařky často zapomínají mezery za tečkami v iniciálách křestních jmen a ve zkratkách. Databázový systém tyto chyby neopravuje a ruční náprava v exportovaném souboru by jistě byla utrpením. Vyřešíme to následujícím makrem. Za zkratkou se může vyskytnout interpunkční znaménko, před nějž nesmíme mezeru vložit. Na řádce 177 si tedy nadefinujeme seznam interpunkčních znamének. Vlastní činnost provede makro `\zw@dot`, jehož definice začíná na řádce 179. Všimněte si, že argument tohoto makra není delimitovaný. Nikdy jím tudíž nemůže být meze. Makro tak řeší další potenciální problém: pokud písařka napíše mezeru mezi tečku a čárku, toto makro ji spolkuje. Jestliže je argumentem znak, který se nevyskytuje v seznamu `\zw@no@dot@space`, přidá se meze. Ta je úmyslně poněkud užší než běžná mezislovní meze. Typografická pravidla nepovolují řádkový zlom uprostřed zkratky a za iniciálou křestního jména. V tomto druhu tiskoviny lze uvedenému pravidlu vyhovět pouze za cenu obřích mezislovních mezer. Proto takový řádkový zlom povolíme, ale s velkou penaltou.

Test nepřítomnosti daného znaku v seznamu interpunkčních znamének provádí makro `\zw@if@not@present` definované na řádce 185. Je to zjednodušená varianta podobného makra z balíku `LTXDOC`. Začíná definicí vnitřního makra `\zw@@in` se třemi parametry, přičemž první parametr je ukončen prvním parametrem makra `\zw@if@not@present`, druhý parametr je neomezený a třetí parametr je ukončen tokenem `\zw@@in`. Vnitřní makro obsahuje pouze neukončenou podmínku `\ifx`, která testuje shodu tokenu `\zw@if@not@present` s druhým parametrem. Při volání makra `\zw@@in` musíme použít `\expandafter`, protože argument `#2`, obsahující seznam interpunkčních znamének, musí do vnitřního makra vstoupit expandovaný. Jestliže testovaný znak, který máme v `#1`, v seznamu není, najde `\zw@@in` celý seznam ve svém prvním parametru, druhý parametr obsahuje pouze token `\zw@if@not@present`, který jsme při volání vnitřního makra přidali, třetí parametr je prázdný a podmínka `\ifx` je splněna. V případě, že testovaný znak je v seznamu obsažen, dostane se do prvního parametru jen část seznamu před příslušným znakem. Znak následující za znakem testovaným bude v druhém parametru. Token `\zw@if@not@present` bude až na konci třetího parametru a podmínka `\ifx` nebude splněna.

Vraťme se ještě k řádce 177. Ve výjimečných případech potřebujeme ručně zabránit automatickému vložení mezery. Proto jsme do seznamu interpunkčních znamének přidali primitiv `\relax`. Záhadně vypadá tečka, která je hned prvním znakem seznamu. Má totiž kategorii 12, ale `\zw@dot` je rozvojem aktivní tečky. Zdálo by se tedy, že tečku s kategorií 12 nemůžeme nikdy ve vstupním proudu

najít. skutečnost je ovšem jiná. Nepočítáme s tím, že by v databázi omylem zůstala zdvojená tečka. Může se však objevit tečka přidaná jiným makrem, ale tato tečka již není aktivní. Bohužel tím vznikne zdvojená tečka, ale takové případy se již na úrovni maker řeší nesnadno (stejně jako pomocí maker nehledáme překlepy a pravopisné chyby).

```

173  {\catcode'\.\active
174  \gdef.{\zw@dot}}
175  \def\ZW@dot{.}
176
177  \def\zw@no@dot@space{.,.;/?!()-\relax}
178
179  \def\zw@dot#1{\ZW@dot
180    \expandafter\zw@if@not@present\expandafter #1\expandafter
181    {\zw@no@dot@space}%
182    \penalty 9999 \hskip .33em plus .1em minus .13em
183    \fi #1}
184
185  \def\zw@if@not@present#1#2{%
186    \def\zw@in##1#1##2##3\zw@in{\ifx\zw@if@not@present##2}%
187    \expandafter\zw@in#2#1\zw@if@not@present\zw@in}

```

Při sazbě potřebujeme často ručně upravit text tak, aby zůstal zachován stránkový rejstřík. Některé záznamy se proto musí vysadit volněji nebo naopak těsněji, někdy je nutno poněkud upravit některé vertikální mezery, případně prodloužit či zkrátit stránku o řádek. Není rozumné zasahovat do souboru, který byl generován automaticky z databáze. Po odladění sazby se totiž může stát, že se v databázi najdou překlepy, které jsou opraveny, soubor exportován znovu a o veškeré úpravy tím přijdeme. Proto makro `\orignum` (viz řádek 166) umožňuje provádět libovolné příkazy na začátku nebo po skončení tisku záznamu. Tyto příkazy se předem definují v hlavním souboru pomocí maker `\after` a `\at`. Jejich definice jsou níže na řádcích 190 a 196.

```

188  % AFTER lecture/poster number
189
190  \def\after{\begingroup \catcode'\ =9 \zw@after}
191  \def\zw@after#1#1{\endgroup \expandafter\def
192    \csname after #1\endcsname}
193
194  % AT lecture/poster number
195
196  \def\at{\begingroup \catcode'\ =9 \zw@at}
197  \def\zw@at#1#1{\endgroup \expandafter\def
198    \csname at #1\endcsname}

```

Nejrafinovanější činnost provádějí následující makra. Začneme pomocnými makry, jež slouží k převodu čísla na půlden. Ta jsou využita během rozvoje makra `\newsect` a dalších. Jejich definice začíná níže na řádku 214. Znovu si změníme kategorie citlivých znaků a po otestování, že se zahajuje nová sekce, nadpis vytiskneme.

```

199 \newcount\zw@cnt
200
201 \newcommand\zw@days[1]{\zw@cnt#1
202   \advance\zw@cnt\m@ne
203   \divide\zw@cnt\tw@
204   \ifcase\zw@cnt Monday\or Tuesday\or Wednesday\or
205     Thursday\or Friday\or Saturday\fi}
206
207 \newcommand\zw@half@day[1]{\ifodd #1 morning\else
208   afternoon\fi}
209
210 \newcommand\zw@next@half[1]{\zw@cnt#1
211   \advance\zw@cnt\@ne
212   \unskip--\number\zw@cnt}
213
214 \newcommand\newsect{\begingroup \zw@cat \zw@@newsect}
215 \newcommand\zw@@newsect[1]{\endgroup\zw@testfirst#1\zw@
216   \zw@typesect#1\zw@}

```

Makro `\newsect`, jehož parametr je zpracován až v `\zw@@newsect`, přebírá ze vstupu komplikovaný text. Jsou jím dvě písmena následovaná číslicí, dvojtečkou a názvem sekce. Příklad vstupního souboru byl již dříve uveden na straně 56. První písmeno označuje typ příspěvku, L je přednáška (lecture), P je vývěska (poster). Druhé písmeno označuje sekci, přičemž X je rezervováno pro plenární přednášky. Číslice definuje půlden. Konverzi na slovní vyjádření provádějí makra `\zw@days` (řádek 201) a `\zw@half@day` (řádek 207). Plenární přednášky nemají jemnější členění. Vývěsky jsou sice dále děleny do sekcí, ale odpovídající nadpisy se vyskytnout nesmí. To by totiž vedlo ke zmatení. Proto makro `\zw@testfirst` dostává celý argument makra `\zw@@newsect` rozkousaný do 5 parametrů a podle typu příspěvku si nadefinuje `\zw@test@it`. Toto pomocné makro je testováno s makrem `\zw@oldsect`, jež je na počátku prázdné. Vlastní tisk hlavičky nové sekce provede makro `\zw@newsect`. Jeho obsahem je pouze cvičení s boxy, linkami a penaltami a použije se v něm `\chilbl` pro vytvoření návěstí pro křížové odkazy. Definice tedy není pro pochopení důležitá a k makru `\chilbl` se vrátíme později. Makro `\plenary@days` (řádek 234), používané při tisku plenárních přednášek, je pro předběžný program definováno tak, aby svůj argument spolkló. Jeho definice je modifikována při rozvoji makra `\final` (viz řádek 129 a následující).

```

217 \def\zw@testfirst#1#2#3#4#5\zw@{%
218   \if #1P%
219     \def\zw@test@it{POSTERS}%
220   \else
221     \if #2X%
222       \def\zw@test@it{PLENARY LECTURES}%
223     \else
224       \def\zw@test@it{#2}%
225     \fi \fi
226   \ifx\zw@test@it\zw@oldsect \else
227     \let\zw@oldsect\zw@test@it \zw@newsect{\zw@test@it}%
228     \if #2X%
229       \addcontentsline{toc}{paragraph}{\zw@test@it}%
230     \fi
231   \fi}%
232 \def\zw@oldsect{}
233
234 \let\plenary@days@gobble

```

Makro `\zw@typesect` je určeno pro tisk nadpisů jemnějšího členění sekcí. Opět provádíme testy typu příspěvku. Zde jistě namítnete, že se makra dala napsat efektivněji. V kódu je ovšem vidět historický vývoj, kdy na modifikace nebylo příliš mnoho času a takovýto přístup byl pohodlnější, než přepisování celých maker s rizikem ztráty řady vnitřních vazeb. S tím také souvisí způsob, jak vytváříme záznamy pro obsah. Pro plenární přednášky se text zapisuje do pomocného souboru již v makru `\zw@testfirst`. Všimněte si, že vždy nastavujeme `\the@open` na určitou pevnou hodnotu. Je to proto, že při řešení problémů při sazbě potřebujeme výjimečně hodnotu jinou. Makro se samo postará o přiřazení původní hodnoty.

```

235 \newbox\zw@chibox
236 \newdimen\zw@chidim
237
238 \def\X@vm{\vskip-\baselineskip}
239
240 \def\zw@typesect#1#2#3:#4\zw@{\vskip 3\baselineskip
241   \let\toc@line\empty
242   \setbox\zw@chibox=\vbox
243   \bgroup \bfseries \fontsize{8}{10}\selectfont
244   \raggedright \hyphenpenalty\@M
245   \if #2X%
246     \aftergroup\X@vm
247   \plenary@days{#3}%

```

```

248 \else
249 \if #1L%
250 #2#3 Lectures\,---\,\zw@days{#3} \zw@half@day{#3}%
251 \xdef\toc@line{#2#3 Lectures}\else
252 \if #1P%
253 P#3 Posters\,---\,\zw@days{#3}%
254 \xdef\toc@line{P#3 Posters}%
255 \xdef\the@lbl##1{\noexpand
256 \chilbl{P#3#1-\noexpand##1}}%
257 \else
258 #2#3#1\,---\,\zw@days{#3} \zw@half@day{#3}\fi
259 \fi\\
260 #4\fi
261 \egroup
262 \dp\zw@chibox=\z@
263 \zw@chidim=\ht\zw@chibox \advance\zw@chidim -2pt
264 \divide\zw@chidim \baselineskip
265 \zw@cnt=\zw@chidim %\advance\zw@cnt\@ne
266 \the@open \def\the@open{\penalty -50}%
267 \vglue \zw@cnt \baselineskip
268 \noindent\box\zw@chibox\nopagebreak
269 \par \clubpenalty\@M
270 \ifx\toc@line\empty \else
271 \addcontentsline{toc}{paragraph}{\toc@line: #4}\fi
272 \if #1P \else \chilbl{#2#3#1}\fi \nobreak }

```

Makro `\chilbl` je použito pro generování návěstí, které se potřebuje v křížových odkazech. Ty se vyskytují v textové části, kterou členové programové komise píšou ručně a odvolávají se v ní na tuto část, která je generována programově. Pro přednášky obsahuje příslušné jméno pouze tři znaky: označení sekce, číslo půldne a typ příspěvku. Vývěsky mají navíc číslo příspěvku. Makro je použito výše na řádcích 256 a 272. Definice, uvedená níže, zabráňuje duplikovanému použití a dále vytvoříme pohodlnější makro místo `\PDFpageref` (varianta `\pageref`, která současně vytváří hypertextový odkaz pro PDF).

```

273 \newcommand\chilbl[1]{\@ifundefined{?chi:#1}{\label{chi:#1}}%
274 \expandafter\gdef\csname ?chi:#1\endcsname{Duplicated}}{}
275 \newcommand\p[1]{\PDFpageref{chi:#1}}

```

Pro vytváření PDF žádná další makra nepotřebujeme. O vše se postará balík PDFHDR.

Vytváření WWW stránek

Původní makra vznikala v době, kdy elektronické komunikace byly jen doménou vědecko-fantastických románů. Proto se vůbec se zveřejňováním informací na WWW nepočítalo. Zpracování na PC pomocí zde popisovaných maker bylo poprvé použito na konferenci, která se konala v roce 1993, zatímco popis HTML verze 2.0 v RFC 1866 pochází z listopadu 1995 a definice protokolu HTTP/1.0 v RFC 1945 dokonce až z května 1996 (HTTP/0.x jsem nenašel). Pokud by makra byla psána znovu v současné době, jistě by se našel vhodnější postup. Za stávající situace byl ovšem problém vyřešen tak, že se vyměnil stylový soubor. Úvodní textová část není totiž shodná s tištěnou verzí a stejně se musí napsat ručně (úpravou holého textu s odstraněnými \TeX ovými sekvencemi). Do HTML se tedy převádí pouze odborný program a autorský rejstřík. Jiná je i struktura. Každá sekce má svoji stránku, informace o vývěskách je také rozdělena podle dnů a obsah s hypertextovými odkazy je vytvářen automaticky.

Hlavní soubor pro generování WWW stránek je velmi jednoduchý:

```
276 \documentclass{article}
277 \usepackage[final]{chi2html-2000}
278 \begin{document}
279 \begin{html}{chisa-scipgm}
280 \input{shemfin}
281 \end{html}
282 \begin{autindex}
283 \input{autind}
284 \end{autindex}
285 \end{document}
```

Podobně jako v tištěné verzi také zde rozlišujeme předběžný a finální program. Příkazy `\input` načítají stejné soubory, z nichž jsme vytvořili tištěnou verzi a PDF. Podívejme se, jak jsme změny výstupu docílili. Uvedeme si skoro celý stylový soubor. Vynecháme makra, která zůstala stejná a některá makra speciální, jejichž význam pro vysvětlení činnosti není zajímavý. Řádky jsou často dlouhé a pro účely zveřejnění ve Zpravodaji musely být opět ručně přelámané.

Začínáme deklarací pomocného čítače a maker, která budeme později často potřebovat. Makro pro přechod na nový řádek sice v žádném z dalších maker nenajdete, ale vyskytuje se na několika místech ve vstupním souboru. Připravíme si též výstupní soubory a makra pro zápis.

```
286 \newcounter{maxpart}
287 \@addtoreset{section}{part}
288 \edef\hash{\expandafter\@gobble\string\#}
289 \def\zw@temp{ }
290
291 \def\{<BR>
```

```

292
293 \newwrite\chi@pg
294 \newwrite\chi@toc
295 \def\chi@wrt#1{\immediate\write#1}
296 \def\protect@wrt#1#2{\begingroup
297   \let\protect\noexpand
298   \chi@wrt#1{#2}\endgroup}

```

Na začátek souboru budeme zapisovat standardní hlavičku. Makro, které bude tuto činnost provádět, potřebuje dva parametry: číslo výstupního souboru a nadpis stránky.

```

299 % Front matter: file title.
300
301 \def\front@matter#1#2.{%
302   \chi@wrt#1{<!DOCTYPE HTML PUBLIC
303     "-//W3C//DTD HTML 3.2//EN">}%
304   \chi@wrt#1{<HTML>}%
305   \chi@wrt#1{<HEAD>}%
306   \chi@wrt#1{<TITLE>CHISA 2000 - #2</TITLE>}%
307   \chi@wrt#1{</HEAD>}%
308   \chi@wrt#1{<BODY bgcolor="\hash ffd16f"
309     text="\hash 000000" link="\hash 1011af"
310     vlink="\hash 1011af" alink="\hash ff322e">}%
311   \chi@wrt#1{<H1>#2</H1>}}

```

Konec stránky bude také standardní. Bude obsahovat odkoky na ostatní stránky. Jen poslední stránka se liší tím, že neobsahuje odkaz na stránku následující.

```

312 % Back matter
313
314 \def\back@matter#1{%
315   \chi@wrt#1{<hr>}%
316   \ifx\chi@lastpage\relax \else
317     \expandafter
318     \ifx\csname Next \arabic{part} page\endcsname\relax
319     \else
320       \chi@wrt#1{<a href=%
321         "@nameuse{Next \arabic{part} page}">%
322         <img border="0">%
323         \chi@wrt#1{src=%
324           "http://hroch486.icpf.cas.cz/icons/forward.gif">}%
325         \chi@wrt#1{Next page</a>}%
326         \chi@wrt#1{<br>}%

```

```

327 \fi
328 \chi@wrt#1{\a href="\chi@fn\chi@lastpage\chi@ext"%
329 <img border="0"%
330 \chi@wrt#1{src=%
331 "http://hroch486.icpf.cas.cz/icons/back.gif">}%
332 \chi@wrt#1{Previous page</a>}%
333 \chi@wrt#1{<br>}%
334 \chi@wrt#1{\a href="chisa-scipgm-org.html">Organisation
335 of scientific program</a>}%
336 \chi@wrt#1{<br>}%
337 \chi@wrt#1{\a href="chisa-scipgm-toc.html">Scientific
338 program, table of contents</a>}%
339 \chi@wrt#1{<br>}%
340 \chi@wrt#1{\a href="chisa-author-index.html">Author
341 index</a>}%
342 \chi@wrt#1{<br>}%
343 \fi
344 \chi@wrt#1{\a href="http://www.chisa.cz/2000/">CHISA 2000
345 home page</a>}%
346 \chi@wrt#1{<hr>}%
347 \chi@wrt#1{This page is maintained by}%
348 \chi@wrt#1{\a href="mailto:wagner@mbox.cesnet.cz">%
349 <nobr>Z. Wagner</nobr>}%
350 \chi@wrt#1{<code>&lt;wagner@
351 mbox.cesnet.cz&gt;</code></a>}%
352 \chi@wrt#1{and was last edited <nobr>on \today.</nobr>}%
353 \chi@wrt#1{</BODY>}%
354 \chi@wrt#1{</HTML>}%
355 \immediate\closeout#1\relax}

```

Kód na konci autorského rejstříku je poněkud odlišný:

```

356 \def\autindex@end#1{%
357 \chi@wrt#1{<hr>}%
358 \chi@wrt#1{\a href="chisa-scipgm-org.html">Organisation
359 of scientific program</a>}%
360 \chi@wrt#1{<br>}%
361 \chi@wrt#1{\a href="chisa-scipgm-toc.html">Scientific
362 program, table of contents</a>}%
363 \chi@wrt#1{<br>}%
364 \chi@wrt#1{\a href="http://www.chisa.cz/2000/">CHISA 2000
365 home page</a>}%
366 \chi@wrt#1{<hr>}%

```

```

367 \chi@wrt#1{This page is maintained by}%
368 \chi@wrt#1{<a href="mailto:wagner@mbox.cesnet.cz">%
369 <nobr>Z. Wagner</nobr>}%
370 \chi@wrt#1{<code>&lt;wagner@%
371 mbox.cesnet.cz&gt;</code></a>}%
372 \chi@wrt#1{and was last edited <nobr>on \today.</nobr>}%
373 \chi@wrt#1{</BODY>}%
374 \chi@wrt#1{</HTML>}%
375 \immediate\closeout#1\relax}

```

Pro zápis HTML souboru si nadefinujeme nové prostředí. V něm nejprve otevřeme nový výstupní soubor, jehož jméno je zadáno jako parametr. V tomto souboru bude obsah s hypertextovými odkazy na jednotlivé stránky. Protože stránky si budou číst lidé z celého světa, kteří mohou mít na svých počítačích nejrůznější kódové stránky a mnozí vůbec nemusí mít font s akcentovanými znaky, raději všechny odstraníme. Uvědomme si, že stránky jsou určeny běžným uživatelům, kteří nesledují vývoj softwarových nástrojů a často mají instalovanu prastarou verzi WWW prohlížeče, která nemusí podporovat UNICODE. Navíc budou WWW stránky umístěny na serveru, k němuž nemám administrátorská práva. Nelze tedy využít MultiViews, což by umožnilo předávat proměnné informace podle toho, co WWW prohlížeč podporuje. Pro sazbu je použit em \TeX a v TCP tabulce je nastavena konverze vstupních znaků na \TeX ové sekvence. Stačí tedy předefinovat několik maker na řádcích 391–396. Podobné definice lze použít i ve spojení s balíkem INPUTENC. Odstraníme také `\hbox`, který mohl být někde využit pro potlačení dělení slova. Na konci prostředí si do pomocného souboru vypišeme hodnotu čítače `maxpart`. Ta bude využita pro nalezení poslední stránky.

```

376 \newenvironment{html}[1]{%
377 \immediate\openout\chi@toc=#1-toc.html
378 \def\chi@fn{#1-}%
379 \setcounter{part}{0}%
380 \def\chi@lastpage{org}%
381 \def\chi@ext{.html}%
382 \frontmatter\chi@toc Scientific program -
383 table of contents.%
384 \chi@wrt\chi@toc{<P><a
385 href="\chi@fn\chi@lastpage\chi@ext">%
386 Organisation of scientific program</a>}%
387 \chi@wrt\chi@toc{<P><a href="chisa-author-index.html">%
388 Author index</a>}%
389 \chi@wrt\chi@toc{<HR><P>}%
390 \def\hbox##1{##1}%

```

```

391 \def\'##1{##1}%
392 \def\v##1{##1}%
393 \def\"##1{##1}%
394 \def\r##1{##1}%
395 \def\c##1{##1}%
396 \def\i{i}%
397 }%
398 {\ifnum\value{part}>0 \back@matter\chi@pg\fi
399 \ifnum\value{subsection}>0 \chi@wrt\chi@toc{</ul>}\fi
400 \ifnum\value{section}>0 \chi@wrt\chi@toc{</ul>}\fi
401 \let\chi@lastpage\relax
402 \back@matter\chi@toc
403 \protect@wrt@mainaux
404 {\protect\setcounter{maxpart}{\arabic{part}}}}

```

Zde je makro pro změnu kategorie vybraných znaků.

```

405 \newcommand\zw@cat{\catcode '\& 12
406 \catcode '\_ 12
407 \catcode '\% 12
408 \catcode '\$ 12
409 \catcode '\# 12}

```

Makra `\num` a `\numtime` musí hlavně vytvořit návěští, na které se budeme odkazovat zejména z autorského rejstříku. Opět máme odlišnou definici pro předběžný a finální program. Tento stylový soubor byl však psán poměrně nedávno již pro $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X} 2_{\epsilon}$, proto je použito mechanismu `\DeclareOption`.

```

410 \def\my@@@test{x}
411
412 \providecommand\num[1]{\chi@a@name{#1}\startit}
413 \def\numtime#1#2#3{\ifx\zw@test@it\my@@@test\else
414 \chi@a@name{#1}\startit\fi}
415
416 \DeclareOption{final}{%
417 \renewcommand\num[1]{\chi@a@name{#1}\startit
418 \chi@wrt\chi@pg{<B>#1</B>}}
419 \def\numtime#1#2#3{\chi@a@name{#1}\startit
420 \chi@wrt\chi@pg{<B>#1 #2.#3</B>}}
421
422 \ProcessOptions
423
424 \def\chi@a@name#1{\chi@wrt\chi@pg{<A name="#1">}}

```

Později budeme potřebovat metodu, jak do pomocného souboru zapsat definici makra. Poslouží nám k tomu `\protect@namedef`. Definované makro musí být globální.

```

425 \DeclareRobustCommand*\protect@namedef[1]{\expandafter
426 \gdef\csname#1\endcsname}

```

Nejprve se podíváme na makro pro zápis hlavičky nové sekce. Toto makro nejprve zavolá `\zw@testfirst`, jehož definice začíná na řádku 429. Pak teprve provede vlastní výstup nadpisu pomocí makra `\zw@typesect`. Definici tohoto makra najdete později na řádce 467. Makro `\zw@testfirst` má opět různé varianty podle typu příspěvku. Pro pozdější zpracování si musíme uschovat nadpis v `\chi@title` a označení sekce v `\zw@test@it`. Toto označení ovšem bude později využito i pro vytváření hypertextových odkazů. Musí být proto převedeno do minusek. U vývěsek a plenárních přednášek obsahuje pevné písmeno, které lze v definici zapsat přímo. Případ přednášek je složitější a správné pořadí expanze vyžaduje použití několika příkazů `\expandafter`. Vidíte je na řádcích 438–440. Na řádce 443 se kontroluje, zda se zahajuje nová sekce, což vyžaduje ukončení aktuální HTML stránky. Uzavření provede makro `\back@matter` a současně se uchovávají informace, které poslouží v dalších makrech k vytváření hypertextových odkazů na právě ukončenou stránku. Zápis nové stránky zahájí makro `\zw@newsect`.

```

427 \newcommand\newsect[1]{\zw@testfirst#1\zw@
428 \zw@typesect#1\zw@}
429 \def\zw@testfirst#1#2#3#4#5\zw@{%
430 \if #1P%
431 \def\zw@test@it{p#3}%
432 \zw@days{#3}\edef\chi@title{Posters #2#3, \zw@d}%
433 \else
434 \if #2X%
435 \def\zw@test@it{x}%
436 \def\chi@title{Plenary lectures}%
437 \else
438 \edef\zw@temp{#2}%
439 \lowercase\expandafter{\expandafter\def
440 \expandafter\zw@test@it\expandafter{\zw@temp}}%
441 \def\chi@title{Lectures #2}%
442 \fi \fi
443 \ifx\zw@test@it\zw@oldsect \else
444 \ifnum\value{part}>0 \back@matter\chi@pg
445 \let\chi@lastpage\zw@oldsect\fi
446 \let\zw@oldsect\zw@test@it
447 \zw@newsect{\zw@test@it}\fi}%
448 \def\zw@oldsect{}

```

Níže uvedené makro `\zw@newsect` nejprve nadefinuje jméno výstupního souboru. Dále se musí postarat o zápis do HTML stránky s obsahem. Musíme uzavřít

odpovídající počet elementů ``, což poznáme podle nenulových hodnot čítačů `section` a `subsection`. Čítač `part` zatím ještě obsahuje pořadové číslo předchozí stránky, zvýšen bude až později. Makro `\zw@xxx`, definované na řádce 454 přiřadí současné jméno souboru k číslu předchozí stránky. Protože výsledek tohoto přiřazení byl vyžadován již dříve na řádce 318 v makru `\back@matter`, nemáme jinou možnost než zapsat hodnotu do pomocného souboru a načíst ji při dalším průchodu. Vytvořili jsme tím podobný mechanismus, jaký L^AT_EX používá pro křížové odkazy. Posun čítače `part`, otevření souboru a zápis hlavičky jsou již triviální.

```

449 \def\zw@newsect#1{%
450   \edef\full@filename{\chi@fn\zw@test@it\chi@ext}%
451   \ifnum\value{subsection}>0 \chi@wrt\chi@toc{</ul>}\fi
452   \ifnum\value{section}>0 \chi@wrt\chi@toc{</ul>}\fi
453   \ifnum\value{part}>0
454     \def\zw@xxx{\protect@namedef{Next \arabic{part} page}%
455               {\full@filename}}%
456     \protect@wrt@\mainaux{\zw@xxx}%
457   \fi
458   \stepcounter{part}%
459   \chi@wrt\chi@toc{<br><a href=%
460     "\chi@fn\zw@test@it\chi@ext">\chi@title</a>}}%
461   \typeout{Opening \full@filename}%
462   \immediate\openout\chi@pg=\full@filename
463   \front@matter\chi@pg \chi@title.%
464   \chi@wrt\chi@pg{<hr>}}

```

Před výkladem makra `\zw@typesect` si znovu občerstvíme význam jeho parametrů. První písmeno je L pro přednášky, P pro vývěsky a X pro plenární přednášky. Druhé písmeno označuje sekci, přičemž vývěsky mají vždy sekci P. Na třetím místě je číslice označující půlden a za dvojtečkou je název sekce. Makra `\zw@days` a `\zw@half@day` jsou nyní předefinována tak, aby výsledek vložila do maker `\zw@d` a `\zw@h`. Makro `\chi@ref` bude obsahovat text použitelný pro hypertextové odkazy uvnitř stránky, `\chi@name` bude užitečné pro vytvoření návěští na stránce pomocí elementu ``. Název si uschováme v `\zw@temp`. Pokud se název liší od předchozího (před prvním použitím je na řádce 465 definováno jako prázdné), musíme opět zapsat odpovídající počet uzavírajících ``. Formát nadpisu se liší podle typu příspěvku a jeho studium ponecháváme čtenářům za domácí úkol.

```

465 \def\chi@sectname{}
466
467 \def\zw@typesect#1#2#3:#4\zw@{\zw@days{#3}\zw@half@day{#3}%
468   \def\chi@ref{\hash#2#3#1}%

```

```

469 \def\chi@name{#2#3#1}%
470 \edef\zw@temp{#4}%
471 \ifx\zw@temp\chi@sectname \else
472 \ifnum\value{subsection}>0 \chi@wrt\chi@toc{</ul>}\fi
473 \ifnum\value{section}=0
474 \chi@wrt\chi@toc{<ul compact>}\fi
475 \stepcounter{section}%
476 \fi
477 \def\ZW@chinum{}%
478 \if #2X%
479 \edef\zw@temp{<h2>\zw@d</h2>}%
480 \else
481 \if #1L%
482 \chi@wrt\chi@pg{<a name="\chi@name"></a>}%
483 \edef\zw@temp{<h2>#2#3 Lectures - \zw@d \zw@h</h2>}
484 \chi@wrt\chi@pg{\zw@temp}%
485 \def\ZW@chinum{#2#3 }%
486 \else
487 \if #1P%
488 \edef\chi@label{Chi Ref \arabic{part}
489 \arabic{section}}%
490 \expandafter\ifx\csname\chi@label\endcsname\relax
491 \else
492 \edef\chi@ref{hash\@nameuse{\chi@label}}%
493 \chi@wrt\chi@pg
494 {<a name="\@nameuse{\chi@label}"></a>}%
495 \expandafter\let\csname\chi@label\endcsname\relax
496 \fi
497 \else
498 \edef\zw@temp{<h2>#2#3#1 - \zw@d \zw@h</h2>}\fi \fi
499 \ifx\zw@temp\chi@sectname \else
500 \chi@wrt\chi@toc{<li><a href="\chi@fn\zw@test@it
501 \chi@ext\chi@ref">\ZW@chinum #4</a>}%
502 \def\chi@subsect{} \let\chi@sectname\zw@temp
503 \fi
504 \edef\zw@temp{<h2>#4</h2>}\fi \chi@wrt\chi@pg{\zw@temp}}

```

Makro `\subsect` slouží k vytváření nadpisů subsekcí. Za zmínku stojí nový způsob využití makra `\chi@ref`, které bylo již naplněno hodnotou na řádku 468. Prostřednictvím makra `\test@P` z něj vybereme druhý a čtvrtý znak a vložíme je do pomocného makra `\zw@p`. zmíněné znaky obsahují typ příspěvku a označení sekce. V případě vývěsek jsou oba znaky shodné, u všech ostatních typů se znaky

liší. Proto na řádce 513 testujeme, zda se jedná o vývěsku a můžeme provést jiné příkazy.

```

505 \def\subsect#1{\def\zw@temp{#1}%
506 \def\test@P##1##2##3##4##5\@chi{\def\zw@p{##2##4}}%
507 \ifx\chi@subsect\zw@temp \else
508 \ifnum\value{subsection}=0
509 \chi@wrt\chi@toc{<ul compact>}\fi
510 \stepcounter{subsection}%
511 \fi
512 \expandafter\test@P\chi@ref\@chi
513 \expandafter\if\zw@p \else
514 \expandafter\ifx\csname chi@name\endcsname\relax \else
515 \chi@wrt\chi@pg{<a name="\chi@name"></a>}%
516 \let\chi@name\relax
517 \fi
518 \fi
519 \edef\chi@label{Chi Ref \arabic{part}
520 \arabic{section}/\arabic{subsection}}%
521 \expandafter\ifx\csname\chi@label\endcsname\relax
522 \else
523 \edef\chi@ref{\hash\@nameuse{\chi@label}}%
524 \chi@wrt\chi@pg{<a name="\@nameuse{\chi@label}"></a>}%
525 \fi
526 \ifx\chi@subsect\zw@temp \else
527 \chi@wrt\chi@toc{<li><a href=%
528 "\chi@fn\zw@test@it\chi@ext\chi@ref">#1</a>}%
529 \let\chi@subsect\zw@temp
530 \fi
531 \chi@wrt\chi@pg{<H3>#1</H3>}}

```

Makra pro zápis jednotlivých záznamů se také liší. Všimněte si, že jsme změnilí též definici standardního makra \-, které mohlo být někde použito pro vyznačení správného dělení. Řádek 541 ověřuje typ příspěvku stejným způsobem, jaký byl již použit na řádce 513. Na řádce 558 začíná poměrně složitý zápis k pomocnému souboru. Zapisují se údaje, které budou v dalším průchodu použity k dopředným hypertextovým odkazům. Zbytek kódu je přehledný a nevyžaduje podrobný komentář.

```

532 \def\main{*}
533 \def\af#1{<sup>#1</sup>}
534 \def\-{ }
535 \def\startit{\chi@wrt\chi@pg{<P>}}
536

```

```

537 \def\orignum[#1]{\ifx\zw@test@it\my@@@test\chi@a@name{#1}\fi
538 \def\chi@aux##1##2##3##4##5@@@chi{\def\chi@{##2##3##4}}%
539 \def\test@P##1##2##3##4##5@@@chi{\def\zw@P{##2##4}}%
540 \expandafter\test@P\chi@ref@@@chi
541 \expandafter\if\zw@P
542 \edef\chi@label{Chi Ref \arabic{part} \arabic{section}}%
543 \edef\chi@sublabel{Chi Ref \arabic{part}
544 \arabic{section}/\arabic{subsection}}%
545 \ifnum\value{subsection}>1
546 \let\chi@label\chi@sublabel
547 \fi
548 \ifnum\value{subsection}=1
549 \expandafter
550 \ifx\csname ZW_TEST \chi@label\endcsname\relax \else
551 \let\chi@label\chi@sublabel
552 \fi
553 \fi
554 \expandafter
555 \ifx\csname ZW_TEST \chi@label\endcsname\relax
556 \expandafter
557 \def\csname ZW_TEST \chi@label\endcsname{Used!}%
558 \expandafter\chi@aux\chi@ref@@@chi
559 \def\zw@P{\protect@namedef{\chi@label}{\chi@-#1}}%
560 \protect@wrt\@mainaux{\zw@P}%
561 \fi
562 \fi
563 \theRefType
564 \chi@wrt\chi@pg{\Naz}%
565 \chi@wrt\chi@pg{<b>\Aut</b>}%
566 \chi@wrt\chi@pg{\Afil}%
567 \chi@wrt\chi@pg{[#1]}%
568 \chi@wrt\chi@pg{</A>}}% end of <A name="...">
569
570 \newcommand\authors{\begingroup \zw@cat \zw@aut}
571 \newcommand\zw@aut[1]{\xdef\Aut{#1}\endgroup}
572 \newcommand\afil{\begingroup \zw@cat \zw@afil }
573 \newcommand\zw@afil[1]{\xdef\Afil{(#1)}\endgroup}
574 \newcommand\nazev{\begingroup \zw@cat \zw@nazev}
575 \newcommand\zw@nazev[1]{\xdef\Naz{#1}\endgroup}
576
577 \newcommand\RefNormal{\def\theRefType{}}\RefNormal
578 \newcommand\RefType[1]{%

```

```

579 \def\theRefType{\edef\zw@temp
580   {<b>\@nameuse
581     {Ref_#1_type}</b>}\chi@wrt\chi@pg{\zw@temp}}

```

Prostředí pro zápis HTML stránky autorského rejstříku je podstatně jednodušší. Nevytváříme v něm žádné sekce. Musíme jen správně interpretovat rejstříková makra a všechny ostatní obraty již byly vysvětleny.

```

582 \newenvironment{autindex}{%
583   \immediate\openout\chi@pg chisa-author-index.html
584   \def\hbox##1{##1}%
585   \def\'##1{##1}%
586   \def\v##1{##1}%
587   \def\"##1{##1}%
588   \def\r##1{##1}%
589   \def\c##1{##1}%
590   \def\i{i}%
591   \front@matter\chi@pg Author index.%
592   \chi@wrt\chi@pg{<P>}
593   \chi@wrt\chi@pg{<A href="\hash A">A</A> |}
594   \chi@wrt\chi@pg{<A href="\hash B">B</A> |}
595   %%%
596   %%%                               Mnoho analogických řádků vynecháno...
597   %%%
598   \chi@wrt\chi@pg{<A href="\hash Y">Y</A> |}
599   \chi@wrt\chi@pg{<A href="\hash Z">Z</A>}
600   \chi@wrt\chi@pg{<HR>}
601   \def\aboveXskip{}\def\belowXskip{}\def\par{}\def\bfseries{}%
602   \def\hfill##1\hfill{\chi@wrt\chi@pg
603     {<H2><A name="##1">##1</A></H2>}}%
604   \def\item##1\pfill{\chi@wrt\chi@pg{<BR>##1}}%
605   \def\normalref##1##2.##3[##4]{\let\@eot\empty
606     \lowercase\expandafter{\expandafter
607       \def\expandafter\sect@num\expandafter{##1}}%
608     \def\chi@refnum{##1##2.##3}%
609     \if##1P\def\chi@ref{\sect@num##2}\else
610       \ifnum ##3=\z@ \def\chi@ref{x}\def\chi@refnum{##4}\else
611         \let\chi@ref\sect@num\fi \fi
612     \chi@wrt\chi@pg{<A href="chisa-scipgm-\chi@ref.html\hash
613       \chi@refnum">[##4]</A>}\@gobble}%
614   \let\mainref\normalref}%
615   {\autindex@end\chi@pg}%

```

Pozorný čtenář už jistě zjistil, že makra pro tvorbu WWW stránek vznikla zkopírováním a úpravou maker pro tisk. Nevýhodou takového postupu je nižší efektivita vytvořených maker. Pokud bychom příslušná makra vytvářeli přímo pro daný účel a nebyli bychom zatíženi existujícím kódem, pravděpodobně by se podařilo napsat makra fungující rychleji. Vzhledem k tomu, že použití tohoto balíku je v zásadě jednoúčelové, není to závažná nevýhoda. Nespornou předností je však skutečnost, že doba, která byla potřebná k vývoji zmíněných maker, byla desetinou času nutného k napsání tohoto článku.

Závěr

Článek demonstroval, jak lze z datábazových výstupů vytvářet různé typy souborů. Přitom není nutno používat vyšší programovací jazyky a změna tiskového formátu je poměrně nenáročná. Práci může ještě více zjednodušit použití moderních značkovacích jazyků, např. SGML/DSSSL nebo XML/XSL.

Literatura

- [1] Z. Wagner *Z \LaTeX u přes PostScript do PDF*. Zpravodaj Československého sdružení uživatelů \TeX u, **9** (1–2), 78–105 (1999).

Summary: Co-operation of a Database with \LaTeX

This article demonstrates advantages of connection of a database system with \LaTeX . It shows how various output formats can easily be generated from the same file exported from a database. Explanations are based upon a project which has long time been used for organization of international CHISA conferences.

Jedna z možností databázového publikování v \LaTeX u

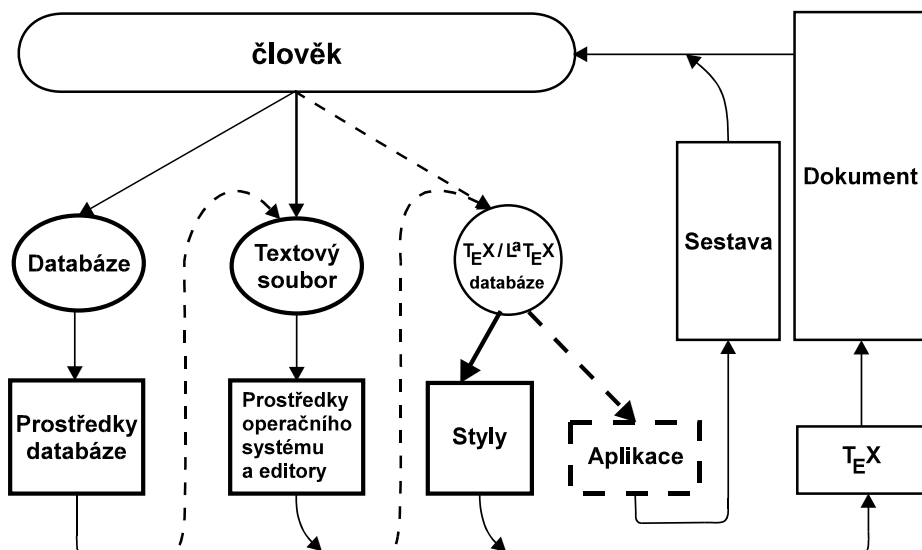
TOMÁŠ HÁLA

Úvod

Stále častěji se setkáváme s případy, kdy připravované dokumenty mají obsahovat výpisy z databázových souborů. Jedná se o zcela obecný problém, který

může být řešen pro různé systémy zpracování textu, pro různé databázové systémy a pro různé operační systémy. Čtenářům tohoto časopisu můžeme připomenout dvě práce na toto téma publikované ve Zpravodaji CSTUG ([6], [4]), což svědčí o tom, že výborné vlastnosti systému $\text{T}_\text{E}\text{X}$ lákají i další autory zabývající se hromadným zpracováním dat.

Následující obrázek znázorňuje různé databázové zdroje, které mohou vstupem do publikačního systému, a různé cesty, kterými se může ubírat zpracování těchto údajů.



Jako zdroje jsou uvedeny:

- *Databázový soubor* určitého databázového systému, který obsahuje vlastní exportní služby umožňující vytvořit sestavu ve formátu jiném, než je vlastní formát.
- *Obecný textový soubor*, jehož vnitřní struktura je předem dána (pro jednoduchost si můžeme tento soubor představit jako tabulku, kde řádky odpovídají záznamům a sloupce položkám). Manipulaci s textovým souborem provádíme zpravidla textovými editory nebo filtry operačního systému.
- *$\text{T}_\text{E}\text{X}/\text{L}^\text{A}\text{T}_\text{E}\text{X}$ ovský databázový soubor¹*, který obsahuje kromě vlastních dat také kontextové značky, odpovídající makropříkazům systémů $\text{T}_\text{E}\text{X}$ a $\text{L}^\text{A}\text{T}_\text{E}\text{X}$. Zde manipulaci s daty zajistíme sestavením a úpravou příslušným maker.

¹Pojem $\text{T}_\text{E}\text{X}/\text{L}^\text{A}\text{T}_\text{E}\text{X}$ ovský databázový soubor je pracovní název. V dalším textu budeme používat zkratku TLDS.

Metoda

Tato metoda byla navržena a poprvé implementována v letech 1994–95 pro soukromé účely autora, který ji používá jednak ve firmě Konvoj, spol. s r. o., pro správu databáze odběratelů, dodavatelů, zboží a služeb, jednak pro osobní účely (evidence vlastních publikací, soukromý telefonní seznam).

Uložení vstupních dat

Při tvorbě výstupních sestav budeme předpokládat, že data jsou uložena v textovém souboru, který má vnitřní strukturu členěnou takto

- Jeden odstavec textu odpovídá jednomu záznamu.
- Všechny záznamy jsou zapsány ve tvaru makropříkazu se shodným identifikátorem.
- Tělem makropříkazu (= záznamu) je posloupnost definic jednotlivých položek, přičemž nezáleží na jejich pořadí.

Předpokládejme tedy, že existuje databáze výrobků a služeb (viz ukázka 1), která obsahuje tyto informace: kód výrobku, název, autor, ISBN, cena a druh titulu.

Všechny informace jsou zachyceny v samostatných položkách. Výjimku tvoří *druh titulu*, tento údaj je implementována v několika položkách, jako jsou turistika, počítače atd. Tyto jednotlivé textové položky doplňují logické údaje.

Ukázka 1 — databáze

```
\z{\kod{0014} \navez{Balada pro banditu} \cena{34,00}  
\dph{5} \zpevnik{+}}
```

```
\z{\kod{1010} \navez{Francie --- Paříž a~Versailles} \cena{31,00}  
\dph{5} \turistika{+} \autor{Jaromír Sopouch, Hana Sopouchová}}
```

```
\z{\kod{0322} \navez{Jak publikovat na počítači} \cena{225,00}  
\dph{5} \pocitace{+} \isbn{80-901475-7-7}}
```

```
\z{\kod{1013} \navez{Kamelot I. --- Zachraňte koně} \cena{20,00}  
\dph{5} \zpevnik{+} \konvoj{+} \isbn{80-85615-02-9}}
```

```
\z{\kod{1062} \navez{Kamelot VII. --- Cesta do ráje} \cena{24,00}  
\dph{5} \zpevnik{+} \isbn{80-85615-52-5}}
```

```
\z{\kod{1029} \navez{Komentovaná chybová hlášení Turbo Pascalu}  
\cena{32,00} \dph{5} \pocitace{+} \isbn{80-85615-19-3} \autor{Jiří
```

Rybička, Tomáš Hála}}

```
\z{\kod{1052} \navez{LaTeX pro začátečníky} \cena{65,00} \autor{Jiří  
Rybička} \dph{5} \pocitace{+} \isbn{80-85615-42-8}}
```

```
\z{\kod{9011} \dph{0} \navez{Poštovné}}
```

```
\z{\kod{9012} \dph{5} \navez{Spěšnina}}
```

```
\z{\kod{9028} \dph{0} \navez{Záloha}}
```

```
\z{\kod{9029} \dph{0} \navez{Vrácení zálohy}}
```

Definice makropříkazů

V ukázce 1 je použita celá řada zatím nikde nedefinovaných makropříkazů. Proto musí existovat tyto úseky zdrojového textu:

- definice položek databáze (ukázka 2),
- definice makropříkazu záznamu (ukázky 3a, 3b),
- definice výstupní sestavy (ukázky 4a, 4b),
- nadřazený úsek (ukázka 5).

Ukázka 2 — definice položek databáze

```
\def\kod#1{\gdef\Kod{#1}}  
\def\navez#1{\gdef\Navez{#1}}  
\def\cena#1{\gdef\Cena{#1}}  
\def\dph#1{\gdef\Dph{#1}}  
\def\zpevník#1{\gdef\Zpevník{#1}}  
\def\pocitace#1{\gdef\Pocitace{#1}}  
\def\turistika#1{\gdef\Turistika{#1}}  
\def\autor#1{\gdef\Autor{#1}}  
\def\isbn#1{\gdef\Isbn{\hbox{ISBN~#1}}}
```

Vidíme, že každá definice makropříkazu položky nastavuje hodnotu jakéhosi globálního makropříkazu. (Tyto globální makropříkazy jsou pojmenovány stejně jako položky, liší se však počátečním velkým písmenem. Tato úprava není podmínkou a činíme tak především pro jednoduchost a přehlednost.) Dále můžeme pozorovat, že globální makropříkazy získávají většinou tutéž hodnotu, jaká je uvedena v parametru položky. Je to nejčastější případ. Poslední řádek ukázky 2 obsahuje definici, obsahují navíc text „ISBN“ a celý údaj je parametrem makropříkazu `\hbox`.

Ukázka 3a — definice makropříkazu záznamu

```
\def\z#1{\init #1 \done}
```

Definici záznamu, tj. souboru položek, obsahuje tři části:

1. inicializaci globálních makropříkazů `\init`,
2. nastavení hodnot a
3. závěrečnou akci, která obsahuje výstup z databáze `\done`.

Inicializace v předchozí ukázce je zcela záměrně nahrazena jedním makropříkazem (`\init`), protože může být provedena odlišně v různých výstupních sestavách.

Ukázka 3b — počáteční definice globálních makropříkazů

```
\def\init{%  
  \gdef\Kod{}           \gdef\Nazev{}           \gdef\Autor{*}  
  \gdef\Dodavatel{*}   \gdef\Dph{}           \gdef\Isbn{*}  
  \gdef\Zpevnik{*}     \gdef\Pocitace{*}     \gdef\Cena{*}  
  \gdef\Pruvodce{*}    \gdef\Turistika{*}  
}
```

Na této ukázce je zajímavé pouze to, že některé položky nabývají prázdné hodnoty a jiné hodnoty neprázdné. Neprázdná hodnota je nezbytná v případě, že výstupní sestava obsahuje podmíněný příkaz (příkaz `\if` apod.).

Ukázka 4a — definice výstupních sestav

Předvedeme si několik sestav. Nejprve zcela jednoduchý výpis všech položek:

```
\def\Soupis{\Kod} \Autor} \Nazev} \Isbn} \Cena}
```

Výstupní sestava obsahuje pět údajů, vypisují se bez úprav za sebou². V případě, že v databázi není některá položka definovaná (viz ukázka 1: např. u položky Poštovné není uveden autor, ISBN atd.), pak údaj je nahrazen implicitní hodnotou dle ukázky 3b.

Ukázka 4b — definice výstupních sestav

Tato sestava je soupisem pouze počítačových publikací, již s drobnými úpravami vzhledu:

```
\def\PocitacoveKnihy{%  
  \if +\Pocitace\Kod} {\it\Autor\/:} {\bf\Nazev}} \\  
  \hphantom{\Kod} }\Isbn} \hfill \Cena\fi}
```

²Otázka pro zvědavé čtenáře: Proč je za názvy položek uvedena prázdná skupina?

Nyní máme již vše připraveno a nezbývá nic jiného, než vyhotovit kýženou výstupní sestavu.

Řídicí modul zde obsahuje (kromě nadpisu) dva povely: nejprve se určí, která výstupní sestava se má použít, a poté se vloží databázový soubor, který obsahuje jednotlivé záznamy, které „volají“ makropříkaz `\done`:

```
\section*{Úplný seznam}
\def\done{\Soupis}
\input{sez.lat}

\section*{Počítačová literatura}
\def\done{\PocitacoveKnihy}
\input{sez.lat}
```

Předložená ukázka vytvoří dokument, který bude obsahovat napřed soupis všech položek databáze (`\Soupis`) a za tím seznam pouze počítačových titulů (`\PocitacoveKnihy`).

Všechny uvedené struktury (definice makropříkazů položek, definice makropříkazu záznamu, definice výstupních sestav, vlastní databáze i řídicí část) mohou být umístěny v jednom souboru nebo rozděleny do několika samostatných souborů podle vkusu a potřeby tvůrce.

Praxe ukázala, že bývá vhodné oddělit vlastní databázi od ostatních částí zdrojového textu, protože se mohou v praxi vyskytnout dokumenty, které vyžadují dvojitý (nebo vícenásobný) průchod databází (viz ukázka 5). Pokud bychom neměli databázi v samostatném souboru, znamenalo by to ji dvakrát (obecně n -krát) vkládat do zdrojového textu, a to se všemi negativními důsledky (opravy v databázi by se musely provádět na více místech, velikost souboru atd.)

Závěr

Zkušenosti s používáním tohoto způsobu tvorby výstupních sestav systémem \TeX , které vycházejí z TLDS, ukazují, že tento způsob je velmi pružný z hlediska rychlosti tvorby výstupních sestav. Rovněž kvalita sestavy je velmi vysoká. Pochopitelnou podmínkou je znalost tvorby makropříkazů v \TeX u, což však není nepřekonatelnou překážkou, neboť existuje dostatek vhodné literatury ([5], [3]).

Drobnou nevýhodou se může jevit za jistých okolností méně komfortní prostředí textových editorů pro zadávání dat do databáze. Existují však další možnosti, jak práci zjednodušit. Jednou z nich je tvorba klávesových maker na klíč, které mohou výrazně editaci zrychlit. Jinou možností je editovat data v komfortnějším prostředí — například pomocí editačních služeb databázových systémů,

pak převést data do TLDS, který je v tomto případě slouží jako standardizované rozhraní pro tvorbu výstupů v $\text{T}_{\text{E}}\text{X}$ u.

Literatura

- [1] HÁLA, T. Zkvalitnění typografické úrovně výstupu z informačního systému. In *Sborník z konference Letní škola informatiky Ruprechtov 1997*. Příloha s. 1–6, 1997.
- [2] OLŠÁK, P. *Typografický systém $\text{T}_{\text{E}}\text{X}$* . 269 s. Praha: CSTUG, 1995.
- [3] OLŠÁK, P. *$\text{T}_{\text{E}}\text{X}$ book naruby*. 468 s. Brno: Konvoj, 1997.
- [4] OLŠÁK, P. Databáze bez databázového programu. Zpravodaj Československého sdružení uživatelů $\text{T}_{\text{E}}\text{X}$ u, **4** (3), 118–125 (1994).
- [5] RYBIČKA, J. *$\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ pro začátečníky*. 169 s. Brno: Konvoj, 1995.
- [6] RYBIČKA, J. Automatizovaná tvorba formulářů pomocí $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ u. Zpravodaj Československého sdružení uživatelů $\text{T}_{\text{E}}\text{X}$ u, **3** (1), 24–27 (1993).

Summary: A possibility of database publishing in $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$

Various documents should often contain listings generated from database files. It is therefore general problem which has to be solved for different text processing systems, database systems and operating systems. One possibility, which is described here, is based on simple text files containing the database records encoded in $\text{T}_{\text{E}}\text{X}$ macros. Experience from using this approach for generation of listings shows that this method is very flexible from the point of view of speed of creation of the output forms. The quality of the listings is also high. The prerequisite is knowledge of the $\text{T}_{\text{E}}\text{X}$ macro language but it is no obstacle because sufficient amount of appropriate literature is available.

Dojmy z 19. bienále grafického designu v Brně

JOSEF SVOBODA

Bienále jsem navštívil poprvé; k tomu, že jsem se do Brna vypravil, mne pohnula témata doprovodných výstav (Adrian Frutiger, Josef Týfa). Výstava o Frutigerovi skončila dříve než celé bienále, už 3. září.

Výstava *Read Me – s Adrianem Frutigerem světem znaků a písmen* spojuje průřez Frutigerovým dílem a úvod do tvorby písma, logotypů, typografie a sazby.

Plocha jedné místnosti je rozdělena mezi oba okruhy asi tak napůl a intenzivně využita. Návštěvník, který sem přišel především kvůli Frutigerovi a méně kvůli základům typografie, tím může být trochu zaskočen. Obě části však stojí za zhlédnutí.

Tři stěny, věnované Frutigerovi, zachycují: 1. chronologický přehled jeho studia a práce, 2. vývoj písem Univers, Frutiger, OCR-A, OCR-B, 3. vizuální informační systémy na letištích a v metru. I tyto tři stěny spolu s tištěným průvodcem, který visí na háčcích pod panely, poskytují vhled do práce tohoto velkého grafika. Autory výstavy jsou Švýcaři Jürg Brühmann a Anja Bodmer. Bohužel ji kazí špatný překlad do češtiny.

Výstava díla *Josefa Týfy* poněkud zklamala: jsou tu vystavena asi všechna písma, která Týfa navrhl, bohužel však nejsou jejich kontury dokonale přesné (asi vinou kopií z fotosazebních předloh); ve vitrínách jsou položeny knihy, které Týfa graficky upravil, a na stěně visí životopis. Nenabízí se tu o mnoho více, než co můžete vidět v knize *Typographia 2*.

Z rozsáhlé expozice vlastního *Bienále* vyberu jen zajímavé návrhy písem. V české expozici se středoštická písmolijna prezentuje svým vzorníkem a především digitální adaptací Preissigovy antikvy (ve spolupráci Otakara Karlase a Františka Štorma). Štorm zmiňuje příbuzenskou linii sahající od Preissigovy antikvy přes Figural a (Růžičkův) Fairfield až k Týfové antikvě. O kvalitě této linie vypovídá, že Figural i Týfova antikva jsou v nabídce ITC.

Velice se mi líbila písma Axela Bertrama (Německo): přechodová antikva Lucinde je současné písmo s nezaměnitelným charakterem, dynamická antikva Rubenau a lomené písmo Salomo jsou vkusně zpracovaná „starobylá písma“.

Viděl jsem návrhy bezpatkových písem pro univerzální použití, která jsou dnes v módě. Bylo jich mnoho a byly si příliš podobné, takže jsem si žádný z návrhů nezapamatoval. Jinou cestou může být sloučení prvků serifových i bezserifových písem v lineární antikvě; takovou vytvořila Yvonne Diedrich z Německa (ITC Diadis). Peter Biľak navrhl kolekci písma Eureka, kde je ze základních tahů odvozena serifová, bezserifová i neproporcionální verze. Jak se mu práce podařila, můžete zhodnotit na www.ui42.sk/peterb.

Jsou tu také velice pěkné hříčky, například Zootype – logotypy, které dokazují, že zvířata mohou žít i uvnitř písmen, z Argentiny. Nebo Stínové písmo, Cycle, Pletar nerad – písma z neobvyklých materiálů od návrhářky Krejčíkové.

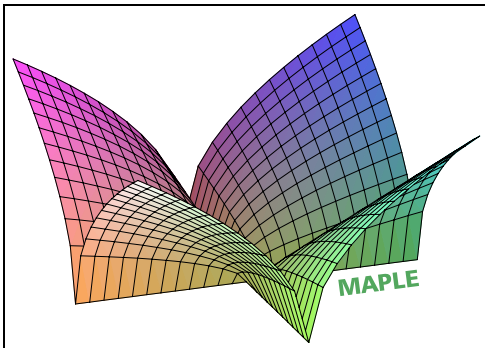
Ve *Studiu Najbrt* bylo možno navštívit výstavu piktografických pajduláků z výroční zprávy. S *Bienále* souvisí také *Grafický design 2000*, výstava českých a slovenských plakátů ve dvoraně obchodního domu. Za zmínku stojí také lávka, po které se sem chodí. Dokládá, jak se dá vytvořit pasáž i v nepříznivých podmínkách.

CD-ROM Matematická analýza s programem Maple:

1. Diferenciální počet funkcí více proměnných

ZUZANA DOŠLÁ, ROMAN PLCH A PETR SOJKA

CD-ROM představuje učební text nového typu využívající možnosti současné výpočetní techniky. Jde o moderní způsob výuky matematické analýzy, kdy prostřednictvím počítačových technologií se student učí matematickou analýzu a naopak. Podnětem k vytvoření CD-ROMu byla potřeba zvýšit geometrickou představivost studentů a zmodernizovat výuku využitím moderních technologií. Jako první partie z matematické analýzy byl vybrán „Diferenciální počet funkcí více proměnných“ a to z těchto dů-



vodů: problémy zde řešené jsou vhodné pro počítačové zpracování a vybrané téma vyžaduje dobrou geometrickou představivost v prostoru. Dalším důležitým důvodem byl i nedostatek zahraničních materiálů k tomuto tématu.

K počítačové realizaci byl vybrán program Maple V pro svoje snadné ovládní a široké rozšíření na vysokých školách v České republice. Vlastní text je uložen ve formátu PDF (Portable Document Format), který je standardem pro elektronickou publikační činnost a je nezávislý na platformě. Kromě jiného umožňuje prostřednictvím křížových odkazů rychle vyhledávat souvislosti napříč celým textem.

CD-ROM je určen pro posluchače odborného studia matematiky, fyziky, informatiky, pro posluchače učitelského studia matematiky a dále všem zájemcům o výuku matematické analýzy s využitím počítače Maple. Materiály zde uvedené vedly k samostatnímu studiu diferenciálních či k přípravě dalších ukulek. Spojení textu, grafiky, stupů by mělo vytvořit profektivnímu zvládnutí probírané problematiky.



čhače učitelského studia matematiky a uživatelům CAS systému ne jsou koncipovány tak, aby němu použití výpočetní techniky počtu funkcí více proměnných materiálů pro podporu výpočetních vstupů a výstředí sloužící k maximálně

CD-ROM je rozdělen do dvou základních částí — na část teoretickou a část praktickou. Praktická část ilustruje využití programu Maple V v diferenciálním počtu funkcí více proměnných. K probírané problematice je zde systémem Maple vytvořena ilustrační grafika a ukázky počítačového řešení příkladů. Teoretická i praktická část jsou úzce svázány prostřednictvím křížových odkazů (po seznámení s teoretickým pojmem si pouhým stiskem tlačítka myši můžeme prohlédnout jeho geometrickou interpretaci a můžeme se seznámit i se způsobem, jakým byla ilustrační grafika vygenerována). Všechny počítačové materiály jsou na CD-ROMu přiloženy. Uživatel CD-ROMu může proto snadno vytvářet analogickou grafiku bez nutnosti zdlouhavého ladění syntaxe příkazů Maplu.

Přílohy obsahují stručný popis programu Maple a dalších systémů počítačové algebry, použitelných k vytváření výukových materiálů. Uvedeny jsou odkazy na doplňující volně šiřitelný software a na neznámější zdroje materiálů pro počítačem podporovanou výuku matematické analýzy na Internetu (volbou odkazu na Internet v PDF souboru se spustí zvolený internetový prohlížeč a zobrazí odpovídající stránku).



CD vzniklo za podpory Fondu rozvoje VŠ v rámci řešení projektu č. 448/1999. Domovská stránka tohoto multimediálního textu je: <http://www.math.muni.cz/~plch/mapm/>. CD-ROM je možno objednat pomocí Internetu na <http://www.math.muni.cz/~plch/mapm/registr.html>. Zdrojový text skript byl pořízen v \LaTeX u. Na CD je z něho text vysázen ve dvou podobách: jedna je určená ke konzumaci na obrazovce, druhá je optimalizovaná pro tisk. K sazbě byly použity standardní balíky z instalace \TeX Live 4 a upravená beta verze styru pdfscreen. Od generování grafiky na žádost odkazy z PDF souborů pomocí MAPLE se nakonec upustilo z finančních/licenčních důvodů.

Zpráva o prezentaci $\mathcal{N}\mathcal{T}\mathcal{S}$ na TUG'2000.

KAREL SKOUPÝ

Čtenáři Zpravodaje již měli možnost seznámit se s existencí projektu $\mathcal{N}\mathcal{T}\mathcal{S}$ na základě článků [1] a [2]. Mým úkolem v tomto projektu je vytvoření reimplementace programu \TeX , která je na jedné straně 100% kompatibilní se „starým dobrým \TeX em“, ale na druhé straně je napsána modulárním způsobem. Cílem této reimplementace je podstatné zvýšení rozšiřitelnosti programu, aby tato

modulární verze mohla sloužit jako základ pro další rozšiřování a experimenty, které povedou především ke zvýšení typografické kvality sázených dokumentů, ale také k uspokojování dalších požadavků ze strany uživatelů.

Ve dnech 12. 8. až 18. 8. 2000 jsem se zúčastnil konference TUG 2000 v Oxfordu, UK. Prezentoval jsem při té příležitosti $\mathcal{N}\mathcal{T}\mathcal{S}$ v jeho aktuální verzi. V té době byla reimplementace již ve velmi pokročilém stádiu, z podstatných částí chybělo pouze zpracování ligatur a kerningů v matematických formulích. Kromě toho zbývalo dokončit ošetření některých chybových situací a další „drobnosti“. Pro samotnou demonstraci programu jsme použili dostatečně přesvědčivý příklad: The TEX book D. Knutha. Ačkoliv DVI výstup nebyl zcela identický s výstupem TEX u, rozdílů nebylo mnoho a vizuálně nebyly příliš patrné (přínejmenším kvůli rozsahu dokumentu).

Pro mne osobně byly více než úspěšná prezentace zajímavé diskuse s dalšími účastníky konference. Měl jsem možnost o vývoji budoucího sázecího systému mluvit s několika TEX ovskými experty, včetně vývojářů dalších „následníků TEX u“. Padly také návrhy na sjednocení zatím poněkud oddělených směrů vývoje, které byly zúčastněnými příznivě přijaty. Ačkoliv $\mathcal{N}\mathcal{T}\mathcal{S}$ ve své současné podobě uživatelům zatím mnoho nového nepřináší, naopak je díky zvolenému implementačnímu jazyku Java mnohonásobně pomalejší než TEX , modulární struktura programu byla velmi oceněna a podle mínění současných rozšiřovatelů TEX u umožní v budoucnu mnohem efektivnější práci.

Při účasti na konferenci mi velmi pomohl ζTUG , který mi uhradil cestovné a část dalších nákladů v celkové výši téměř 20000 Kč. Touto cestou ζTUG u za sebe i $\mathcal{N}\mathcal{T}\mathcal{S}$ tým za jeho sponzorství srdečně děkuji.

Odkazy:

- 1 Hans Hagen: *Some $\mathcal{N}\mathcal{T}\mathcal{S}$ thoughts*. Zpravodaj Československého sdružení uživatelů TEX u, **9** (3), 109–115 (1999).
- 2 Karel Skoupý: *$\mathcal{N}\mathcal{T}\mathcal{S}$: nový sázecí systém*. Zpravodaj Československého sdružení uživatelů TEX u, **9** (3), 115–131 (1999).

Karel Skoupý
skoupy@fi.muni.cz

Zpravodaj Československého sdružení uživatelů T_EXu

ISSN 1211-6661

Vydalo: Československé sdružení uživatelů T_EXu
vlastním nákladem jako interní publikaci

Obálka: Antonín Strejc

Počet výtisků: 800

Uzávěrka: 11. listopadu 2000

Odpovědný redaktor: Zdeněk Wagner

Tisk: Viaprint, Václavské nám. 17, 1. schodiště,
110 00 Praha 1, mobil 0603 814 629

Distribuce: KONVOJ, spol. s r. o., Berkova 22, 612 00 Brno,
tel. 05-740233

Adresa: ČSTUG, c/o FI MU, Botanická 68a, 602 00 Brno

fax: 05-412 125 68

e-mail: cstug@cstug.cz

Zřízené poštovní aliasy sdružení ČSTUG:

bulletin@cstug.cz, zpravodaj@cstug.cz

korespondence ohledně Zpravodaje sdružení

board@cstug.cz

korespondence členům výboru

cstug@cstug.cz, president@cstug.cz

korespondence předsedovi sdružení

cstug-members@cstug.cz

korespondence členům sdružení

cstug-faq@cstug.cz

řešené otázky s odpověďmi navrhané k zařazení do dokumentu ČSFAQ

secretary@cstug.cz, orders@cstug.cz

korespondence administrativní síle sdružení, objednávky CD-ROM

bookorders@cstug.cz

objednávky tištěné T_EXové literatury na dobírku

ftp server sdružení:

<ftp://ftp.cstug.cz/>

www server sdružení:

<http://www.cstug.cz/>

Podávání novinových zásilek povoleno Českou poštou, s.p. OZJM Ředitelství
v Brně č.j. P/2-1183/97 ze dne 11. 3. 1997.