Píšeme veliké nápisy v TEXu

Tento článek je určen pro uživatele s „průměrnými“ znalostmi o TEXu a se žádnými znalostmi o METAFONTu. Ti zkušenější si určitě už dávno poradili s problémem zvětšování písma do libovolné velikosti, přičemž pravděpodobně museli překonat několik počátečních neúspěchů při prvním kontaktu s programem METAFONT. Chtěl bych ukázat, že nyní může i nezasvěcený uživatel tento program snadno použít, aniž by narazil na podobné problémy. Nová instalace CSTEXu umožňuje snadné a bezproblémové zacházení s METAFONTem a přitom není potřeba ani příliš proniknout do podrobností programu.

Netrvá příliš dlouho, kdy nový uživatel TEXu začne požadovat větší písmo, než je v instalacích obvykle k sehnání. Většinou takový uživatel pracuje v LATEXu a použije příkaz `\Huge`. Výsledek se mu zdá příliš malý a proto zkusí příkaz `\HUGE`. V tuto chvíli ovšem už nedostane žádný výsledek, ale pouze chybové hlášení, že příkaz není definován. Začátečník by mohl nabýt dojmu, že zde je mez TEXu a větší písmena prostě nelze získat. To samozřejmě není pravda.

TEX samotný pracuje pouze s neúplnou informací o fontu – bere na vědomí jen rozměry jednotlivých znaků, případně vzdálenosti mezi dvojicemi znaků, nebo požadavek k sazbě ligatury. Tyto informace jsou uloženy v souboru s příponou `.tfm`. Jednotlivé tvary znaků v bitmapové formě načítají až použité DVI-ovladače (např. prohlížeč nebo ovladač tiskárny). Tato druhá část údajů o fontu bývá uložena v souborech s příponou `.pk`, případně celé skupiny fontů jsou shromážděny do knihovního souboru s příponou `.fli`. Zatímco soubor `.tfm` je v instalaci použit jen jednou, bitmapové reprezentace tohoto fontu se většinou vyskytují v instalacích vícekrát různě zvětšené (většinou každé další zvětšení je 1.2 násobek předchozího).

Z uvedeného vyplývá, že pro TEX není problém pronásobit načtené údaje o fontech libovolným koeficientem. Tato možnost samozřejmě existuje a za chvíli si ukážeme, jak se to dělá. Na druhé straně, zvolíte-li pro zvětšení fontu nestandardní koeficient, vznikne v `.dvi` souboru odkaz na zvětšený font, pro který v instalaci neexistuje bitmapová podoba. Ovladače proto budou mít problémy. Tyto problémy odpadají, máme-li instalovaný METAFONT, program, který načítá textové soubory, v nichž jsou popsány jistým jazykem jednotlivé tahy, z nichž se skládají tvary všech znaků ve fontu (tzv. zdrojové texty fontů) a vytváří bitové mapy fontů. Vstupními údaji pro METAFONT jsou samozřejmě také údaje o použitém výstupním zařízení (rozlišovací schopnost) a použité zvětšení fontu (libovolný koeficient).

Po tomto „úvodu“ (odborníci prominou) můžeme přistoupit k vlastnímu experimentování. Popíši možnosti, jak lze problém řešit za pomocí nové instalace CSTEXu. Co k tomu potřebujeme? Mít instalovaný TEX z nové instalace. Není-li zatím instalovaný METAFONT, lze jej doinstalovat z instalačních disket CSTEX-1 a CSTEX-4. Přitom potřebujeme aspoň 3 MB volného místa na disku.

Dále si připravíme konfigurační soubor `texcfg.bat` třeba s tímto obsahem

```
echo ***** Automaticke generovani fontu. *****
set DVIDRV=%DVIDRV% /fb /fm:hplaser
set SCROPT=@scr300.cnf
set MFJOBOPT=/1
```

Zde ve druhém řádku napíšeme `hplaser`, pokud chceme konečný produkt našeho snažení vytisknout na laserové tiskárně, v jiném případě použijeme slovo `lqlores`, `lqhires`, resp. `epsonfx` pro jehličkovou tiskárnu s rozlišením 180, 360, resp. 240 dpi. Tím jsme METAFONTu sdělili typ našeho výstupního zařízení. Na třetím řádku požadujeme, aby prohlížeč pracoval s takovým rozlišením fontů, které odpovídá našemu konečnému výstupnímu zařízení (zde laserová tiskárna s 300 dpi).

V posledním řádku oznamujeme typ použitého METAFONTu. `/1` znamená, že se použije `mf186.exe` a `/3` způsobí vyvolání programu `mf386.exe`. Třetí alternativou, programem `sbmf.exe`, se nyní nebudeme zabývat, protože situace je poněkud komplikovanější.

Pokud máme správně připravený konfigurační soubor `texcfg.bat` a *je přítomný v aktuálním adresáři*, můžeme spustit systém TEX například příkazem

```
latex napis
```

Při startu systému se na obrazovce mimo jiné objeví echo z našeho konfiguračního souboru, což potvrzuje, že konfigurace proběhla.

Napišme nyní v LATEXu tento zdrojový text:

```
\documentstyle{article}
\pagestyle{empty}
\begin{document}
\begin{center}
\bf V~této místnosti \\ nekouříme \\
a \\ nepoužíváme \\ kradený software.
\end{center}
\end{document}
```

Po „přeložení“ LaTeXem se na výsledné dílo můžeme podívat pomocí `View`. Zatím můžeme v klidu „odladit“ vzhled celé zprávy v malém. Požadavek na zvětšení ještě není uveden, takže zatím nemáme žádné problémy.

Pokud jsme se vzhledem zprávy zcela spokojeni, rozhodneme se, kolikrát ji potřebujeme zvětšit. Při těchto úvahách jsme omezeni velikostí papíru, který je schopna přijmout naše tiskárna. Zpráva samozřejmě může být na papíře A4 otočena např. o 90 stupňů.

Například jsme se rozhodli pro šestinásobné zvětšení. Připíšeme tedy před příkaz `\documentstyle` požadavek na zvětšení

```
\mag=6000
```

Číslo 1000 znamená zvětšení jedna ku jedné a v rámci tohoto koeficientu můžeme uvést libovolné celé číslo pro zvětšení či zmenšení. Upozorňujeme, že `\mag` není totéž, co `\magnification`. Druhý příkaz v LaTeXu nefunguje a v ostatních formátech zvětšuje všechny rozměry až na rozměr tiskového zrcadla. To by mohlo vést ke zcela jinému řádkovému zlomu, a to v našem případě nechceme. První příkaz (změna hodnoty vnitřního registru) přitom zvětší zcela všechny rozměry, včetně rozměrů tiskového zrcadla.

Překlad proběhne bez problémů (řekli jsme si, že TeX neřeší problémy s bitmapami). Pokud nyní vyvoláme prohlížeč, objeví se zpráva:

```
Warning 1209: do you want to call MFjob to generate
              1 missing font now ?
Type Y, N or ?:
```

Na tuto otázku odpovíme Y (ano, chceme vygenerovat bitmapu nenalezeného fontu) a pak už jen asi 3 minuty (podle rychlosti počítače) zíráme na obrazovku, kde se vypisují zprávy o vygenerovaných písmenech a nakonec, aniž bychom cokoli zmáčkli, znovu naskočí prohlížeč.

Ovšem ejhle! Prohlížeč nezobrazí vůbec nic – jen zcela prázdnou stránku. To nás nesmí odradit; raději hledejme příčinu tohoto „drobného“ nedostatku. Zaměřme se na referenční bod stránky, který je 1 in (2.54 cm) od horního a levého okraje papíru. Pronásobme vzdálenost nezvětšeného textu od tohoto bodu šestkrát a uvědomme si, že na místě zůstává pouze referenční bod a vše ostatní se zvětšuje a proto i posunuje. V našem případě je po zvětšení text zcela mimo papír A4.

Přidejme do dokumentu *za* příkaz `\documentstyle` požadavek na menší šířku tiskového zrcadla. Vyjadřujeme se v nezvětšených jednotkách. Uvedené hodnoty jsou výsledkem experimentu.

```
\textwidth=6cm \oddsidemargin=-1cm \topmargin=-1cm
```

Po novém překladu a spuštění prohlížeče se už žádný font negeneruje a navíc vidíme část našeho textu. Celý se na stránku A4 na šířku nevejde a proto jej budeme rotovat o 90 stupňů. Přejdeme tedy do nabídky `Print` a tam zvolíme položku `Edit-Options`. V editoru napíšeme pouze hodnotu parametru `tr`, který určuje rotaci.

```
/tr1
```

Když nyní spustíme položku `PreView xxxDPI`, kde `xxx` odpovídá rozlišení tiskárny, pak konečně uvidíme na stránce A4 text celý a otočený. Pokud se nám to tak líbí, pak volba položky pro tisk nám přinese kýžený výsledek.

Bohužel, ukázku tohoto výsledku nemůžeme do bulletinu zařadit, protože se sem prostě nevejde. Aby čtenář získal aspoň představu o kvalitě písma, uvedeme zde z ukázky jedno slovo.

nekouříme

Nově vygenerovaný font se automaticky zařadil do struktury adresářů v instalaci. Takové věci je možno dělat doma na soukromé instalaci TEXu. Generovat si svoje velikosti fontů na veřejných instalacích nemůže každý uživatel. Tyto fonty mají totiž jednorázové použití a je nesmyslné, aby zůstávaly v instalaci natrvalo. Kdyby si každý uživatel vygeneroval font a nechal ho uložený ve veřejné instalaci, disková kapacita jakkoli velkého disku by za chvíli nestačila. Proto bývají veřejné instalace zabezpečené proti zápisu (read only), nicméně i zde lze použít zmíněnou metodu generování fontů. Stačí, když je tato instalace vytvořena z nově distribuovaných disket $\mathcal{C}\mathcal{S}$TEX a obsahuje METAFONT a zdrojové texty fontů. V konfiguračním souboru `texcfg.bat` je nutno změnit SCROPT následovně

```
set SCROPT=@scr300.cnf /pf:@f.pk
```

Tato volba způsobí, že pokud generujeme font při volání prohlížeče `View`, pak se výsledek (soubor s příponou `.pk`) uloží do uživatelského adresáře, takže programy nemají snahu nic ukládat do read-only částí instalace. Přitom všechna ostatní nastavení ovladačů, tj. `PreView` i tisky, dokáží tento `.pk` soubor v uživatelském adresáři najít a použít. Uživatel po skončení práce sám rozhodne, jak naloží se souborem `.pk` ve svém adresáři. Například jej může archivovat a někdy později použít.

Pozor. Tento způsob ukládání fontů má jednu zásadní nevýhodu. Nejsou zde rozlišovány velikosti. Znamená to, že pokud vyhledávací proces nenajde font v instalaci (kde jsou rozlišovány velikosti), pak mu stačí, že v uživatelském adresáři najde font s příslušným názvem a na velikost nehledí. To může vést ke kuriózním výsledkům, pokud je např. v závěru použito zařízení s jinou rozlišovací schopností.

Dále samozřejmě nelze v aktuálním adresáři uchovávat dvě bitmapy téhož fontu a v různé velikosti. V takovém případě je nejvhodnější z pozice uživatele vytvořit si vlastní uživatelský konfigurační soubor `scr300.cnf` a `hplj.cnf`, které, pokud jsou přítomné a aktuálním adresáři, mají přednost před systémovými. V těchto souborech pak uživatel přepíše volbu `/pf` tak, aby se vytvořila při generování fontů struktura adresářů se soubory `.pk` tam, kam má přístup k zápisu.

Na tomto místě bych chtěl uvést doporučení pro TEX-administrátory. Je vhodné přepsat systémové konfigurační soubory `*.cnf` tak, aby prvá alternativa parametru `/pf` obsahovala adresáře přístupné všem uživatelům k zápisu. Tam se budou hromadit nestandardní fonty, které používají uživatelé. TEX-administrátor může jednou za čas stáhnout tyto fonty do společných knihoven, případně některé, u nichž je patrné jednorázové použití, může vymazat.

Je třeba si uvědomit, že tato „snadnost“ použití METAFONTu spočívá v tom, že Mattesovy ovladače lze nakonfigurovat tak, aby při nenalezené bitmapě fontu dokázaly samy vyvolat METAFONT s potřebnými parametry prostřednictvím programu MFJOB, jehož autorem je také Eberhard Mattes. Jako uživatelé vůbec nemusíme vědět, jak se METAFONT vlastně spouští. Nemusíme ani použít nabídku s názvem `METAFONT`, která je také zahrnuta do systému. Tato nabídka tam je zařazena pro tvůrčí činnost s METAFONTem, především pro tvorbu obrázků, nebo pro návrh nových a krásnějších písem. Můžete také deformovat tvary běžně používaných fontů libovolným způsobem. To vše ovšem vyžaduje aspoň trochu umět jazyk METAFONTu. O možnostech v této nabídce si povíme někdy příště.

Nyní si ještě uvedeme, jak lze míchat použití fontů nezvětšených s našimi zvětšenými. Například tento článek je celý napsaný v základní velikosti, tj. hodnota `\mag` je rovna 1000. Přitom je zde použita ukázka slova „nekouříme" v poněkud jiné velikosti. Nelze proto měnit parametr `\mag`. Proto je v článku na začátku dokumentu napsáno:

```
\font\obrovsky=csbx10 scaled 6000
```

a řádek s ukázkou má tvar

```
\centerline{\obrovsky nekouříme}
```

nebo v LaTeXu

```
\begin{center}
{\obrovsky nekouříme}
\end{center}
```

Toto řešení ovšem vyžaduje na uživateli, aby znal názvy použitých fontů. Zde `csbx10` je CS-font s diakritikou odvozený z `cmbx10`, což je Computer Modern font bold extended at 10pt, tedy polotučný řez, použitý při příkazu `\bf`. Pokud bychom ale místo slova `csbx10` napsali `cmbx10`, dostaneme na výstupu pouze zkomoleninu „nekoume" (příslušně zvětšenou), a v `.log` souboru najdeme zprávu:

```
Missing character: There is no ř in font cmbx10!
Missing character: There is no í in font cmbx10!
```

Nemáme-li CS-font, musíme použít originální `cmbx10`, ale pak je nutné přepsat slovo `nekouříme` na sekvenci `nekouříme`.

V prvém případě, když jsme zvětšovali celý dokument parametrem `\mag`, jsme se nemuseli starat o názvy použitých fontů. Přitom byl skutečně použit font `csbx10` místo `cmbx10`, protože jsme použili novou instalaci CSTeXu, kde je tato věc zajištěna při generování formátu.

Nyní už tedy vidíme rozdíl mezi parametrem `\mag`, který působí globálně, a zvětšením pouze jediného fontu.

Kdybych změnil parametr `\mag` ve svém článku globálně, tak by při závěrečné redakci bulletinu došlo při zpracování k chybě

```
! Incompatible magnification (6000);
```

Tento parametr je totiž přenášen na jediné místo do `.dvi` souboru při tisku první strany a pak je kontrolován při ukončení formátování. Ostatní údaje v `.dvi` zůstávají nezměněny. Příslušné místo, kam je hodnota parametru `\mag` zanesena, je čteno až ovladačem, který pak pronásobí všechny rozměrové údaje v `.dvi` požadovaným koeficientem.

Hodnota `\mag` navíc při formátování ovlivní přepočet rozměrů, které mají prefix `true` a to tak, že je pronásobí převrácenou hodnotou koeficientu. Tím se liší její použití od jinak analogického způsobu zvětšení dokumentu až na úrovni ovladače, tj. parametrem `/m`. Přítomnost parametru `/m` způsobí, že ovladač zcela ignoruje hodnotu `\mag` v `.dvi`. Rozměry s prefixem `true` ve zdrojovém textu jsou nyní postiženy také deformací, protože do `.dvi` souboru se tento prefix nepřenáší.

Závěrem ještě poznámku k omezení TEXu a METAFONTu. Maximální hodnota parametru `\mag` je 32768. Tím se nám podaří zvětšit běžné desetibodové písmo na asi 12 cm. Chceme-li větší písmena, musíme udělat zásah do zdrojového textu fontu. Pak ale narazíme na omezení METAFONTu. Maximální rozměr bitmapy, který je tento program schopen vygenerovat je 4096 pixelů od referenčního počátku všemi směry. Tj. maximální výška a šířka je 8192 pixelů v ideálním případě, když máme referenční bod uprostřed. V případě písma je obvyklé mít referenční bod vlevo dole a proto je nutné počítat s číslem 4096. Toto číslo odpovídá při rozlišení 300 dpi velikosti 34 cm, tedy písmeno může být větší, než je rozměr strany A4. Na druhé straně při rozlišení osvitových jednotek (1200 dpi) už máme maximální písmeno velké jen 8 cm. Také nesmíme zapomenout na obecně známé omezení TEXu, že největší rozměr, který lze uložit do `.dvi` je 16384 pt, tedy asi 5.75 m.

Je tedy vidět, že TEX ani METAFONT není určen pro generování písmen na transparenty do prvomájových průvodů. Nic na tom nezmění ani ta skutečnost, že tento článek píši právě v den, kdy se dříve tyto průvody konaly.

1.5.1993 Petr Olšák