

OD NUL A OBČASNÝCH JEDNIČEK KE GENEROVANÝM KNIHÁM FONTÍKŮ A KANDŽÍKŮ

STRÍŽ, Pavel, (CZ)

Abstrakt. Článek je náhledem do světa generovaných knih náhledů písem. Ať už písem s rozsahem 128 či 256 znaků (nespisovně sedmi- a osmibitáčky) nebo písem a jejich glyfů používaných v sazbě ČJKV (čínština, japonština, korejština, vietnamština; nespisovně šestnáctibitáčky). Smyslem článku není poukázat na všechny nuance téměř ročního projektu, spíš má čtenáři zamotat hlavu tím, cože to v tom $\text{T}_{\text{E}}\text{X}$ u, jeho přátelích a ve svobodném software, lze všechno připravit. Čtyři vygenerované knihy vydal nakladatel Martin Stříž.

1 Úvod aneb o jednom snu

Když jsem byl žák končící povinnou školní docházkou (začátek devadesátých let minulého století), tak jsme jako děcka řádili na osmibitáčích. Záhy přichází osobní počítače s mikroprocesorem, operačním systémem DOS a chvíli na to s grafickým uživatelským rozhraním. Tehdy to byl Microsoft Windows 3.11. Éra šestnáctibitáček. Zde jsem se poprvé setkal s editorem T602, AmiPro a Mapou znaků. Srdíčko žáka zaplesalo, že by bylo hezké vidět všechny znaky z nabízených písem. Po dvaceti letech se sen stal skutečností, viz další strana.

Autorský záměr bylo vypsát všechny glyfy (kresby znaků) z vybraných písem, odstranění opakujících se znaků, optimalizovat rozložení do obdélníku, a samozřejmě, vše automatizovaně s vektorovými výstupy. Navíc s nějakou zajímavou úvodní stránkou a rejstříkem.

2 $\text{T}_{\text{E}}\text{X}$

Typografický systém $\text{T}_{\text{E}}\text{X}$ je v mnoha ohledech komplexní nástroj. Je to dílo z pohledu typografie, programování, matematiky i možnosti automatizace. Základem práce u písem jsou formáty PFB (kresba znaků) a TFM (metriky). Po instalaci $\text{T}_{\text{E}}\text{X}$ ové distribuce (např. pomocí $\text{T}_{\text{E}}\text{X}$ Live) je řada písem připravená k ostrému použití. Zde naše bádání začneme.

AntykwTypeormanske-Regulair @ TeX Collection

Přehledy písmaček kůl upel dšabcké odč.
The quick brown fox jumps over a lazy dog.
This sentence contains every letter of the alphabet.
Přehled písmaček kůl upel dšabcké odč.
The quick brown fox jumps over a lazy dog.
This sentence contains every letter of the alphabet.

AntykwTypeormanske-Cond-Bold @ TeX Collection

Přehled písmaček kůl upel dšabcké odč.
The quick brown fox jumps over a lazy dog.
This sentence contains every letter of the alphabet.
Přehled písmaček kůl upel dšabcké odč.
The quick brown fox jumps over a lazy dog.
This sentence contains every letter of the alphabet.

AntykwTypeormanske-Regulair @ TeX Collection

Přehled písmaček kůl upel dšabcké odč.
The quick brown fox jumps over a lazy dog.
This sentence contains every letter of the alphabet.
Přehled písmaček kůl upel dšabcké odč.
The quick brown fox jumps over a lazy dog.
This sentence contains every letter of the alphabet.

AntykwTypeormanske-Cond-Bold @ TeX Collection

Přehled písmaček kůl upel dšabcké odč.
The quick brown fox jumps over a lazy dog.
This sentence contains every letter of the alphabet.
Přehled písmaček kůl upel dšabcké odč.
The quick brown fox jumps over a lazy dog.
This sentence contains every letter of the alphabet.

AntykwTypeormanske-Regulair @ TeX Collection

Přehled písmaček kůl upel dšabcké odč.
The quick brown fox jumps over a lazy dog.
This sentence contains every letter of the alphabet.
Přehled písmaček kůl upel dšabcké odč.
The quick brown fox jumps over a lazy dog.
This sentence contains every letter of the alphabet.

AntykwTypeormanske-Cond-Bold @ TeX Collection

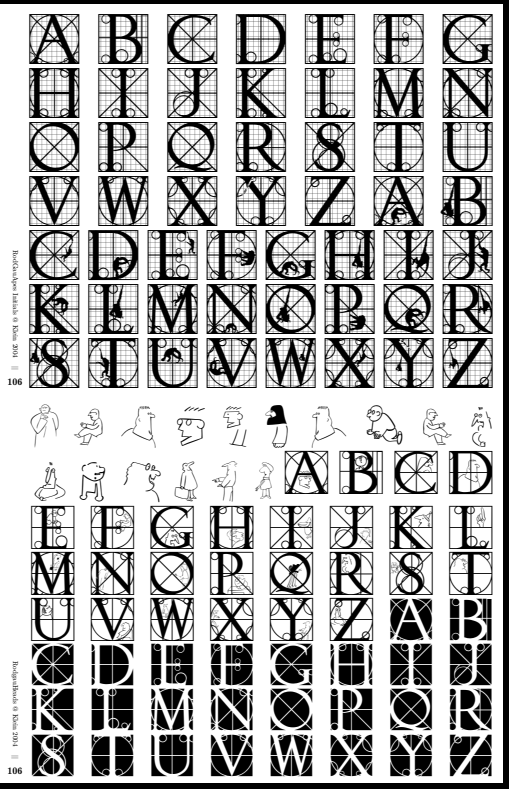
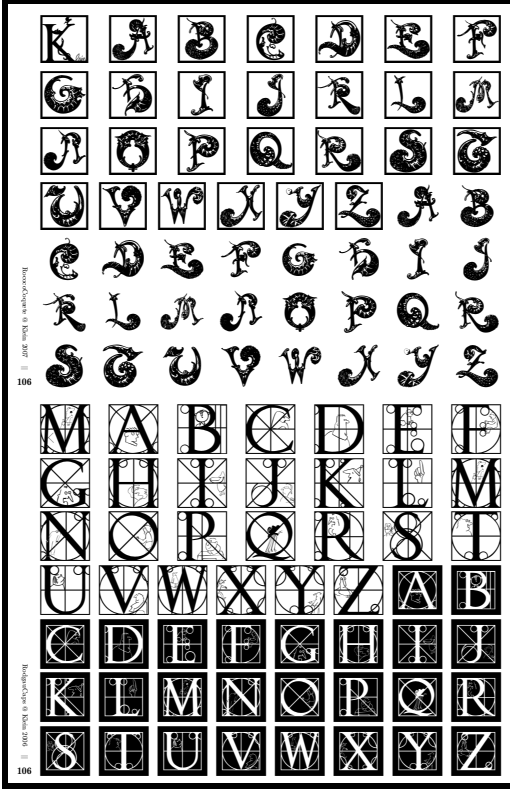
Přehled písmaček kůl upel dšabcké odč.
The quick brown fox jumps over a lazy dog.
This sentence contains every letter of the alphabet.
Přehled písmaček kůl upel dšabcké odč.
The quick brown fox jumps over a lazy dog.
This sentence contains every letter of the alphabet.

AntykwTypeormanske-Regulair @ TeX Collection

Přehled písmaček kůl upel dšabcké odč.
The quick brown fox jumps over a lazy dog.
This sentence contains every letter of the alphabet.
Přehled písmaček kůl upel dšabcké odč.
The quick brown fox jumps over a lazy dog.
This sentence contains every letter of the alphabet.

AntykwTypeormanske-Cond-Bold @ TeX Collection

Přehled písmaček kůl upel dšabcké odč.
The quick brown fox jumps over a lazy dog.
This sentence contains every letter of the alphabet.
Přehled písmaček kůl upel dšabcké odč.
The quick brown fox jumps over a lazy dog.
This sentence contains every letter of the alphabet.



Tři designy z prvním tří knih. Výsledek snah a řady příjemných experimentů.

2.1 Výpis znaku

Vezměme si písmeno »P« od slova Pavel. To můžeme zapsat přímo z klávesnice. Také můžeme využít informaci o pozici znaku v písmu, konkrétně »P« je 80_{10} . Přeloženo do lidské řeči: v ASCII tabulce najdeme písmeno »P« na pozici 80, což je číslo zapsané v desítkové soustavě. Relativně snadno můžeme získat osmičkovou a šestnáctkovou reprezentaci takového znaku. V T_EXu použijeme příkaz `\char` či stříškovou (bohužel ne střížkovou) konvenci.

```
\char'120 \char80 \char"50 ^^50
```

Získáváme: PPPP. Pokud tedy nevíme jak nebo neumíme vygenerovat znaky přímo, můžeme použít tuto zkratku. Při použití cyklu místo konstanty, nyní hodnota 80_{10} , použijeme proměnnou, jak si ukážeme na této a další straně.

2.2 Výpis všech znaků či glyfů

Dáme si první náročnější úkol. Budeme chtít vypsát všechny dostupné znaky používaného písma v T_EXu. Vezmeme si například T_EXové písmo `csr10`. Rychlý náhled v podobě tabulky v PDF získáme makrem `testfont`, tedy zapsáním:

```
pdftex testfont ←
csr10 ←
\table\bye ←
```

Ze začátku je to nezvyk, ale budeme-li chvíli výstup zkoumat, najdeme tam osmičkovou i šestnáctkovou reprezentaci znaků. Také vidíme, že maximum je 256 znaků. Nevidíme ani kerningy. Jedná se o přibližování a oddalování párů znaků, k tomu je potřeba zobrazit například souvislý text. Také není těžké si představit, že při tisících fontů (to je průměrná galerie) by byl při tisku náhled jednoho písma na stránku extrémně finančně náročný. Zkusíme si proto vlastní výpis.

2.3 Definovaný výpis

Výhodou definovaného výpisu je, že máme plnou kontrolou nad rozsahem vypisovaných znaků. Lze si přizpůsobit velikost písma, řádkový proklad, případně si navrhnout dosti specifický design. Můžeme si kontrolovaně vypisovat páry znaků kvůli kerningu. Na ukázkou zvolíme netradiční vektorové písmo s japonskou slabičnou abecedou hiragana.

```
\font\hira=dmjhira at 9.245pt % Načtení písma a jeho aktivace.
\newcount\citac \citac=-1 % Inicializace čítače.
{\sloppy % Nepřetékání řádků.
\loop % Zahájení cyklu.
\advance\citac by 1 % Navýšení čítače o jedničku.
{\hira\char\citac}% % Výpis specifického znaku nebo glyfu.
```

```

\discretionary{}{}{}%           % Lze zalomit řádek za každým glyfem.
\ifnum\citac<255\repeat         % Výpis celého rozsahu 0-255.
\par}                            % Konec skupiny pro \sloppy.
\normalfont\selectfont         % Zpět do základního písma.

```

Výstupem získáváme:

ああいうええおおかがきぎくくげげごさざしじすずせせそそただちちつづつてととなに
ぬねのはははひびびふぶふへべへほほまみむめもややゆゆよよらりるれろわわみゑをん

Můžeme si před či za glyfem vypsát jeho číselnou reprezentaci užitím `\the\citac`. V té chvíli však zjistíme, že u většiny písem není všech 256 pozic obsazených. To při jisté jednotě zobrazování činí nemalé potíže. Jednou máme před sebou základní sadu 26 znaků, vedle toho písma s diakritickými a speciálními znaky, jednou s a jednou bez číslic atd.

Naši ukázkou si rozšíříme. Abychom nemuseli měnit rozsah ručně (u hiragany by byl vhodný od 1 po 83), lze si změnit a otestovat délku nebo výšku znaku. Pokud je délka nulová, čítač a neexistující glyf se nezobrazí.

```

\fontsize{6.6}{8}\selectfont % Úprava velikosti písma a prokladu.
\font\kata=dmjkata at 7.15pt % Načtení písma a jeho aktivace.
\newcount\citac \citac=-1 % Inicializace čítače.
{\sloppy \hfuzz=0pt         % Formátování odstavce.
\pretolerance=50 \tolerance=50 % Tolerance přetečení u řádků.
\fontdimen3\font=0pt \fontdimen4\font=0pt \fontdimen7\font=0pt
\loop                       % Rozpal mezery a zahájení cyklu.
\advance\citac by 1         % Navýšení čítače o jedničku.
\setbox0=\hbox{\kata\char\citac} % Změříme glyf, \wd0, \ht0 a \dp0.
\setbox1=\hbox{}%         % Změříme si prázdný box.
\ifnum\wd0=\wd1\else       % Je délka znaku nulová?
\textttf{\ifnum\citac<10 0\fi % Přidání nuly před cifry 0-9.
\the\citac.\copy0 }%      % Výpis specifického znaku nebo glyfu.
\discretionary{}{}{} %   % Lze zalomit řádek za každým glyfem.
\fi%                       % Konec podmínky \ifnum.
\ifnum\citac<255\repeat    % Výpis celého rozsahu 0-255.
\par} % Ukončení odstavce a uzavření lokální skupiny.
\normalsize\selectfont    % Zpět do základního písma.

```

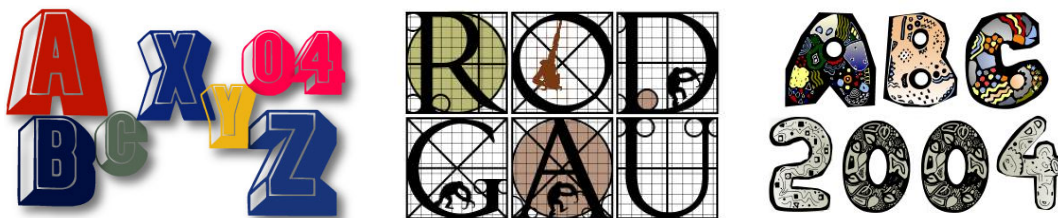
Výstupem dostáváme očíslovaný seznam glyfů katakany bez prázdných pozic.

01. ア 02. ア 03. イ 04. イ 05. ウ 06. ウ 07. エ 08. エ 09. オ 10. オ 11. カ 12. カ 13. キ 14. キ 15. ク 16. ク 17. ケ 18. ケ
19. コ 20. コ 21. サ 22. サ 23. シ 24. シ 25. ス 26. ス 27. セ 28. セ 29. ソ 30. ソ 31. タ 32. タ 33. チ 34. チ 35. ツ 36. ツ
37. ツ 38. テ 39. テ 40. ト 41. ト 42. ナ 43. ニ 44. ニ 45. ネ 46. ノ 47. ハ 48. ハ 49. バ 50. ヒ 51. ビ 52. ビ 53. フ 54. プ
55. プ 56. ヘ 57. ヘ 58. ベ 59. ホ 60. ボ 61. ボ 62. マ 63. ミ 64. ム 65. メ 66. モ 67. ヤ 68. ヤ 69. ヨ 70. ヨ 71. ヨ 72. ヨ
73. ラ 74. リ 75. ル 76. レ 77. ロ 78. ワ 79. フ 80. キ 81. コ 82. コ 83. シ 84. ヴ 85. カ 86. ケ

Nyní vidíme, že kdybychom chtěli zajistit design do cca definovaného poměru obdélníku pro více písem, tak to není triviální úloha. První čtyři řádky jsou v pořádku, pátý je příliš krátký. Jedna z možností je postupné zvětšování/zmenšování velikosti písma, např. metodou půlení intervalu, ovšem i tak zůstává nevýhodou nedotažení posledního řádku.

2.4 Presentování písem

Grafici a designéři si prezentují svá písma nejrůznějšími způsoby. Zde je několik náhledů na Kleinova písma. Jedná se o TTF DoubleFaces, RodGauApes Initials a FilledABC z roku 2004. U prvního zmíněného náhledu je potřeba podotknout, že Manfred Klein byl mezi úplně prvními, kteří začali vyvíjet 3D písma, chceme-li písma s 3D efekty.



2.5 Pangram

Řada písmolijen prezentuje svá písma za použití pangramů (kontrola existence znaků, náhled písma při různých velikostech) a souvislých textů (pohled na ligatury a kerning párů).

```
\def\pCZ{Vřešticí přišery se dožadovaly úplně čerstvých řízečků.}
\def\pSK{Päťtýždňové vlčatá nervózne štekajú na mōjho ďatľa v trní.}
```

Nastavit specifickou velikost písma a řádkového prokladu lze příkazem `\fontsize`, konkrétně např. `\fontsize{13.5}{15}\selectfont`. Jedná se o body (pt).

Ukážeme si výpis pangramu se změnou velikosti písma s/bez změny řádkového prokladu. Nejjednodušší ukázka se změnou velikosti písma je:

```
\newcount\vyska \vyska=2
\loop
\advance\vyska by 1
%\fontsize{\vyska}{\vyska}\selectfont % Ke srovnání.
\font\mujfont=csr10 at \vyska pt \mujfont
\par{\pCZ\ Písmo csr10 ve velikosti \the\vyska pt.}
\ifnum\vyska<10\repeat
\normalsize\selectfont
```

Výstupem získáváme:

Vřešticí přišery se dožadovaly úplně čerstvých řízečků. Písmo csr10 ve velikosti 3pt.

Vřešticí přišery se dožadovaly úplně čerstvých řízečků. Písmo csr10 ve velikosti 4pt.

Vřešticí přišery se dožadovaly úplně čerstvých řízečků. Písmo csr10 ve velikosti 5pt.

Vřešticí přišery se dožadovaly úplně čerstvých řízečků. Písmo csr10 ve velikosti 6pt.

Vřešticí přišery se dožadovaly úplně čerstvých řízečků. Písmo csr10 ve velikosti 7pt.

Vřešticí přišery se dožadovaly úplně čerstvých řízečků. Písmo csr10 ve velikosti 8pt.

Vřešticí příšery se dožadovaly úplně čerstvých řízečků. Písmo csr10 ve velikosti 9pt.

Vřešticí příšery se dožadovaly úplně čerstvých řízečků. Písmo csr10 ve velikosti 10pt.

Následuje ukázka se zásahem do řádkového prokladu:

```
\newdimen\vyska \vyska=2pt
\newdimen\proklad \proklad=0pt
\loop
\advance\vyska by 1pt
\proklad=1.22\vyska
\fontsize{\vyska}{\proklad}\selectfont % Dochází k substituci.
\font\mujfont=csr10 at \vyska \mujfont % Přepsání výšky.
\par\makebox[3cm][l]{\the\vyska/\the\proklad}\pSK
\ifdim\vyska<10pt\repeat
\normalsize\selectfont
```

Výstupem dostáváme:

3.0pt/3.60pt	Přítýždňové vlčatá nervózne štekajú na môjho datľa v tñní.
4.0pt/4.88pt	Päťtýždňové vlčatá nervózne štekajú na môjho datľa v tñní.
5.0pt/6.1pt	Päťtýždňové vlčatá nervózne štekajú na môjho datľa v tñní.
6.0pt/7.32pt	Päťtýždňové vlčatá nervózne štekajú na môjho datľa v tñní.
7.0pt/8.54001pt	Päťtýždňové vlčatá nervózne štekajú na môjho datľa v tñní.
8.0pt/9.76001pt	Päťtýždňové vlčatá nervózne štekajú na môjho datľa v tñní.
9.0pt/10.98001pt	Päťtýždňové vlčatá nervózne štekajú na môjho datľa v tñní.
10.0pt/12.20001pt	Päťtýždňové vlčatá nervózne štekajú na môjho datľa v tñní.

2.6 Balíček shapepar

Pokud chceme vysázet výpis znaků do definovaného obdélníku, můžeme použít \TeX ový balíček shapepar. Do preambule přidáme `\usepackage{shapepar}`, dokumentaci nejrychleji získáme zapsáním v příkazové řádce: `texdoc shapepar` ←.

```
\newcount\znak
\font\mujfont=wncyr10 at 7.5pt
\def\vypis{
  \znak=-1
  \loop
  \advance\znak by 1
  {\mujfont\char\znak}%
  \discretionary{}{}{}%
  \ifnum\znak<255\repeat
  } % Konec příkazu \vypis.
\begin{minipage}{0.3\textwidth} \heartpar{\vypis}\end{minipage}
\begin{minipage}{0.4\textwidth} \starpar{\vypis}\end{minipage}
\begin{minipage}{0.3\textwidth}\hexagonpar{\vypis}\end{minipage}
```

Výstupem získáváme všechny znaky (a srdíčko navíc) písma wncyr10 ve třech verzích:

НЬЛЦЭИ ЄЂТНЬЛ џѡієћћЮЖЙЇЄVΘSЯюжйївєся “!” Ъ“% ‘ ‘()* *Љ,-./0123456789:;<« 1»?˘ АБЦДЕФГХИJKLMНОП ЧРСТУВШШЫЗ[“] Ъ Ъ ‘а б цдефгхијкљмнопчрсту вщшыз—Њњъ ♥	ЉЦ ЭІЄЂ ТНЬЛЦЭІЄ ћћЮЖЙЇЄVΘSЯюжйївєся“!” Ъ“% ‘ ‘()* *Љ, -./0123456789:;<1»?˘ АБЦДЕФГ ХИJKLMНОПЧРСТ УВШШЫЗ[“] Ъ Ъ ‘абцдефгхијкљмнопч рстувш шыз— № Ъ Ъ	НЬЛЦЭІЄЂТНЬ љџієћћЮЖЙЇЄVΘ СЯюжйївєся“!” Ъ“% ‘ ‘() *Љ,-./0123456789:;<1»?˘ АБ ЦДЕФГХИJKLMНОПЧРС ТУВШШЫЗ[“] Ъ Ъ ‘абц дефгхијкљмнопчрст увщшыз—Њњъ
--	---	--

Zjistit automatizovaně zalomení řádků je zdánlivě náročný úkol. Autor použil přístup, že si výsledné PDF převedl na text a z něj potřebné detaily zjistil. Speciální znaky byly opět převedeny na T_EXové sekvence s příkazem `\char`.

```
pdftotext -nopgrbrk -raw -eol unix vstup.pdf vystup.txt ←
```

Řešení zdánlivě nejtěžšího problému, jak zajistit zaplnění celého definovaného obdélníku, spočívalo jen v nalezení největší velikosti písma. Stačilo kontrolovat, zda-li je výstupní PDF na jedné straně, nebo již přeteklo na dvě strany. Mezi každým znakem se nastaví příkaz, který pruží. Tím se zajistí optimální rozložení po délce. Podobně se nechají pružit řádkové proklady. V běžné sazečské praxi se používají příkazy `\hfill` a `\vfill`. Výstupní PDF lze ořezat o ochranný prostor nástrojem `pdfcrop` nebo zjištěním bounding boxu, např. přes GhostScript. Ukázky zdrojových kódů jsou bohužel nad rámec tohoto článku.

2.7 Rotace

Praktickým problémem především u rozsáhlých písmových projektů Manfreda Kleina byla jiná situace. Byly to opakující se kresby a kresby otočené zrcadlením. U náhledů totiž stačí jedna vizuální varianta a uvolněný prostor lze využít na zvětšení písma. Připomeneme, že rotace a zrcadlení nejsou již náročné operace na úrovni T_EXu samotného. Následující kód zpracuje znak »Ž« z příjmení Stríž a »y« kvůli testování hloubky znaků, to vše za pomoci balíčku `graphicx` (dokumentaci lze získat opět přes `texdoc graphicx` ←):

```
\setbox0=\hbox{Žy} \unhcopy0\ --\
\reflectbox{\unhcopy0}\ --\
\rotatebox[origin=c]{180}{\unhcopy0}\ --\
\reflectbox{\rotatebox[origin=c]{180}{\unhcopy0}}
```

Existuje nespočet možností rotace, ale s těmito třemi se lze setkat u písem na úrovni TTF/OTF nejčastěji. Zásadní je samozřejmě zrcadlové překlopení (2. možnost). Ukázky jsou následující: $\text{Žy} - \text{v}\check{\text{Z}} - \text{^}\check{\text{Z}} - \text{v}\check{\text{Z}} - \text{^}\check{\text{Z}}$. Občas bylo možné se setkat i s inverzí, zešikmením a dalšími transformacemi kreseb znaků.

2.8 Identita a podobnost znaků

V našem projektu bylo možné vyřadit stejné nebo velmi podobné znaky. To by \TeX uměl (především Lua \TeX), ale není to triviální problém. Na vyhledávání duplicitních a otočených kreseb lze použít program FontForge, <http://fontforge.sourceforge.net/>.

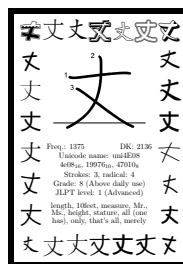
FontForge je poměrně unikátní projekt, který umí pracovat s písmy v grafickém režimu, dávkově i jako knihovna Pythonu. Má svůj vlastní programovací jazyk a automatizovat s ním operace s písmy není výrazný problém.

V článku nám na ukázky již prostor nezůstal, proto autor srdečně zve na přednášku na konferenci OSSConf2010 (začátek července), a také na čtvrté setkání $\text{CON}\text{T}\text{E}\text{X}\text{T}$ istů a konferenci TEX perience 2010 (září; mlýn Brejlov u Prahy; konference jsou pod záštitou Ministerstva školství, mládeže a tělovýchovy České republiky). Akce společně organizují $\text{C}\text{S}\text{T}\text{U}\text{G}$ a FaME UTB ve Zlíně.

3 Zajímavosti kolem knih



I	IwonaMedium-Regular Baf!	138
J	Jana Skrivana Bold Baf!	139
	Jana Skrivana BoldSlant Baf!	139
	Jana Skrivana Regular Baf!	140
	Jana Skrivana Rslant Baf!	140
	Jana Skrivana Rslant Baf!	141
	Jana Skrivana Slant Baf!	141
	JFJungleRock Baf!	355
	Unicode-BoldItalic Baf!	142
	UnicodeBold Baf!	142
	UnicodeItalic Baf!	143
	UnicodeRegular Baf!	143



3.1 Úvodní listy

S úvodními listy jsme si „vyhráli“. Svět typografie má rád hladkost a uhlazenost, matematici a statistici mají rádi naopak náhodu a chaos. Jak to spojit dohromady?

Vyzkoušeli jsme tuto cestu. Sazba náhodně volených znaků byla klasicky v řádcích, ale jednotlivé znaky byly náhodně otáčeny a posouvány tam či zpět z aktuální pozice. Výsledkem se podařilo rozumné rozložení šedi na papíru se zahrnutým efektem náhodnosti.

Ukážeme si jeden z možných principů, jak z dostupné sady znaků zkontrolovaných a vypsaných FontForgem náhodně (s vrácením i bez vrácení) vybírat. Zdrojový kód jednoduché ukázky může vypadat takto:

```
#!/bin/bash
glyfy=znaky.txt glyfydoc=ztemp.txt # Pomocné soubory.
echo -e "A\nB\nC\nD\nE\nF" >$glyfy # Pracovní sada.
RANDOM=100 tahusop=15 tahubez=7
pocet=$(cat $glyfy | sed -n '$=' ) # Počet řádků.
if [ $pocet -lt $tahubez ]; then
```



```


echo -e "Opravuji počet tahů bez navracení na $pocet.\n"
let tahubez=$pocet
fi
function volba {
let "vyber=$RANDOM % $pocet + 1" # Jedna až počet řádků.
radek=$(cat $glyfy | sed -n "$vyber p") # Vypiš řádek.
echo $i. volba je $vyber. řádek. Obsahem je $radek.
}
# Výběr s navracením.
for i in $(seq 1 1 $tahasop); do # Tahy s opakováním.
volba
done; echo # Konec for cyklu.
# Výběr bez vracení.
for i in $(seq 1 1 $tahubez); do # Tahy bez opakování.
volba
cat $glyfy | sed "$vyber d" >$glyfydoc # Odebere daný řádek.
# cat $glyfy | sed "/$radek/d" >$glyfydoc # Alternativa.
cp $glyfydoc $glyfy
echo "Výpis zbytku souboru:"; cat $glyfy; echo
let pocet=$pocet-1
done
rm -f $glyfy $glyfydoc

```

Prezentovaný princip je prostý.

V textovém souboru máme výpisy znaků, vždy jeden na každém řádku (písmeno nebo to může být příkaz). Spočteme celkový počet řádků takového souboru a z něj volíme řádek. Poté buď následuje zásah do souboru v podobě vymazání voleného řádku (výběr bez navracení), nebo nikoliv (výběr s vracením).

3.2 Rejstřík

Tohle byla jedna ze zapeklitějších situací, které autor na sebe ušil, ale i to se podařilo vyřešit. Charakteristické je, že u každého písma je několik znaků písmem vysázených (Baf!), ty charakterizují malá písmena (»af«; mohlo by být i »á«, zástupce diakritických znaků), verzálky (»B«) a zástupce speciálních znaků (»!«). Takto čtenář hned vidí, co lze v písmu přibližně očekávat. Ještě obecnější by bylo »Báf2×«, kde je i číslice a matematický symbol. Inspirace vznikla z rejstříku prezentovaného v závěru dokumentace `texdoc symbols` .

Další charakteristiky rejstříku jsou, že je generovaný strana po straně kvůli velkému počtu písem v každé knize, poté jsou strany spojeny při kompletaci. Každá strana začíná třídícím znakem. Kapitoly knihy jsou odlišeny řezem písma čísla stránky.

U kandžíků jsou rejstříky navíc předgenerovány za pomoci nástroje Bash.

JLPT	椰	178	諄	339	上	25
Level 1	楓	179	諒	339	部	368
(by freq.)	汀	193	謁	341	東	167
	汰	194	賦	349	者	287
	洵	198	迪	361	党	45
	滉	206	遵	366	地	78
且	濤	210	醇	370	合	63
丙	濫	212	采	371	市	111
丞	熙	217	銑	375	業	179

3.3 Kompletace

U PostScriptových souborů se používaly nástroje `psselect` a `psbook`. U zpracování PDF lze použít L^AT_EXový balíček `pdfpages`. Máme-li například PDF o osmi stranách, můžeme jej přesázet na dvě strany v režimu 2×2 takto:

```
\documentclass[a4paper]{article}
\usepackage{pdfpages}
\begin{document}
\includepdf [pages={-},nup=2x2,frame]{vstup.pdf}
%\includepdf [pages={-},pagecommand={Hello World!}]{vstup.pdf}
\end{document}
```

Výhodou zůstává, že se lze vrátit zpátky do sazby vkládané stránky nepovinným parametrem `pagecommand`. V ukázce si znak procenta (komentování řádku) z pátého řádku přesuňte na začátek řádku čtvrtého.

3.4 X_YL^AT_EX

Písma pro tři knihy fontíků byla získána konverzí do TFM (soubor metrik) a PFB (kresba glyfů) nástroji `afm2tfm`. V rámci testů byly využity nástroje `tftopl`, `pltotf` a `vptovf`. Samozřejmě nástroje `Bash`, `Sed`, `Cat` či `Tr` byly na denním pořádku, především u zpracování kanjidic Jima Breena, <http://www.csse.monash.edu.au/~jwb/kanjidic.html>.

Jakmile jsme čelili více než 256 znakům v jednom písmu, vyzkoušeli jsme X_YL^AT_EX. Ten umí přímou práci s kódováním UTF-8. Umí zobrazit jeden znak několika způsoby, viz článek pravděpodobně v *openMagazinu* 5/2010.

Můžeme pracovat s názvem glyfu, včetně názvu `.notdef` a glyfů s indexem nula. Takto lze zjistit počet definovaných glyfů přes různá písma a dle toho upravit sazbu. V navrženém rozvržení nevypadalo hezky, když byly například přítomny tři kandžiky v celém okraji.

Rozmístění objektů bylo zrealizováno prostředím `picture` a načtení více než devíti parametrů v jednom T_EXovém příkazu bylo zrealizováno standardně přes vnořování příkazů. Jeden z následovníků T_EXu, tzv. LuaT_EX, umí tuto možnost ještě elegantněji, ale o tomto rozšíření T_EXu snad až někdy jindy.

3.5 Další zajímavosti

Za zmínku stojí několik maličkostí.

- První kniha obsahuje 703, druhá 2632 a třetí 2741 různých písem a jejich řezů.
- Na stahování dostupných písem z Internetu byl na všechny možné i nemožné způsoby použit program wget.
- Bitmapou byla vložena jen písma, která byla záměrně autorsky vytvořena tak, aby přesáhla technické možnosti písmových formátů.
- Při výrobě knih se zpracovávají TTF, OTF, PFB i DFONT formáty v jednom konverzním běhu díky programu FontForge.
- V knize kandžíků bylo představeno 68 znaků a znamének latinky, 197 speciální glyfů spolu s hiraganou a katakanou a 6376 kandžíků.
- Jako doplněk poslední knihy vznikla postupnými výběry a natažením 2230 kandžíků přes balíček pdfpages metrová housenka.
- Na tisk byly předány PDF ve velikostech 3 MB (housenka), 101 MB (kandžíci), 179 MB (fontíci 1), 255 MB (fontíci 2) a 706 MB (fontíci 3).
- Váha knih je 1,6 kg (kandžíci), 2,2 kg (fontíci 1), 2,3 kg (fontíci 2) a 3,7 kg (fontíci 3).

4 Závěrem snění do budoucna

Na knihy by snad mohly navázat další dva díly, opět s trochu jiným designem. První představující projekt www.ceskefonty.cz s různými transformacemi písem a druhý díl s náhledy 10.000 písem a jejich řezů galerie www.ultimatefontdownload.com, která je dostupná za cenově přijatelných 19,99 USD.

Lze se těšit na knihu čínských znaků, kterých je víc než sto tisíc? Možná, možná až bude UTF-16 na denním pořádku.

Autor se s vámi rozloučí náhledem na jedno z Kleinových písem, kde je vidět, že pomocné a pracovní nákresy (většinou nad rozsahem úvodních 256 znaků) musely být z náhledů odstraněny. Spodní obrázek je zvětšenina znaku »A« ze zmíněného písma z horního obrázku na další straně.

Kontaktní adresa

Ing. Pavel STRÍŽ, Ph.D.

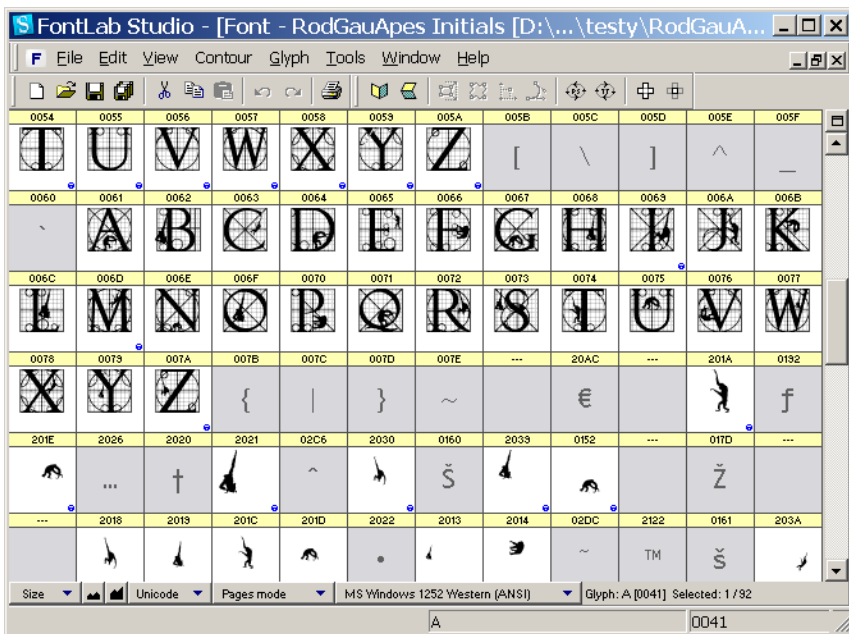
Ústav statistiky a kvantitativních metod

Fakulta managementu a ekonomiky

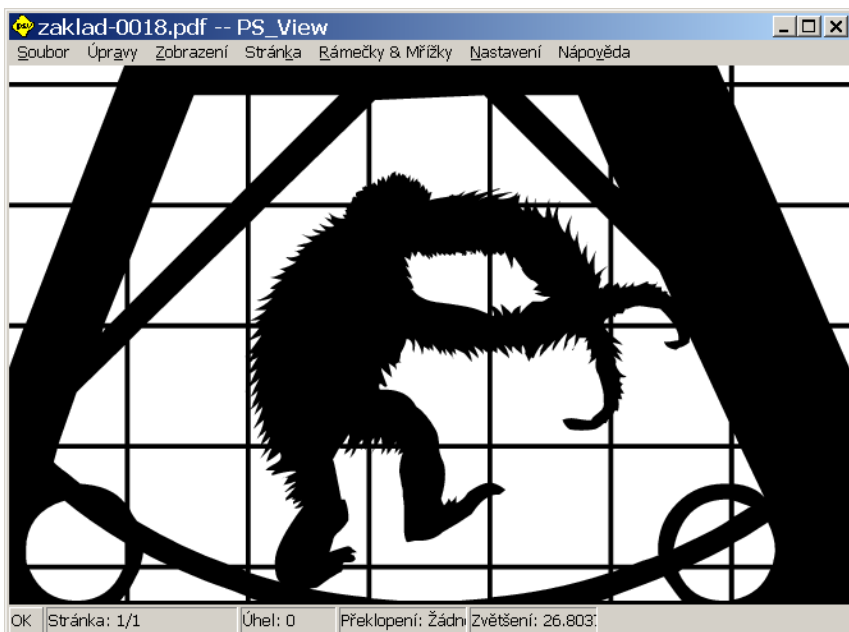
Univerzita Tomáše Bati ve Zlíně

Mostní 5139, CZ-760 01 Zlín

striz@fame.utb.cz



Náhled na písmo RodGauApes Initials od Manfreda Kleina, 2004.
První reakce autora článku byla: „Sakra! Co je to tam za fleky?“



Oblíbený náhled autora v programu PS_View při výrazném zvětšení (nyní 26,8×). Glyph U+0061 v plné kráse.