Zpravodaj Československého sdružení uživatelů TeXu

Petr Sojka; Michal Růžička

Publikování z jednoho zdroje v odlišných formátech pro různá výstupní zařízení

Zpravodaj Československého sdružení uživatelů TeXu, Vol. 18 (2008), No. 3, 116–129

Persistent URL: http://dml.cz/dmlcz/150054

Publikování z jednoho zdroje v odlišných formátech pro různá výstupní zařízení

Petr Sojka, Michal Růžička

TEX je tradičně používán jako autorský nástroj pro publikování vědeckých textů a učebnic. V dnešní době jsou z mnoha důvodů čtenáři vyžadovány elektronické publikace souběžně vydávané nejen ve webovém formátu, ale i ve formě pro prohlížení na obrazovce. Tento článek se zabývá způsoby publikování z jednoho zdrojového textu autorsky pořizovaného ve formátu LATEX a ukazuje příklady několika skript publikovaných touto cestou. Zvláštní důraz je přitom kladen na webové dokumenty generované buď do HTML nebo XHTML s matematickými výrazy převedenými do jazyka MathML.

Zmíněno je také „on-the-fly“ generování dokumentů s JBIG2 komprimovanými obrázky pro potřeby digitální matematické knihovny projektu DML-CZ.

Motivace

Discover the outer logic of the typography in the inner logic of the text.
–Robert Bringhurst [4]

Dokumenty předávající informace mají svůj *obsah* a *formu*. Formu (vzhled) by měl zohledňovat *design*, který by měl užívat grafické prostředky konzistentně. Možnosti formy dokumentu jsou vymezeny *výstupním zařízením* (papír, LCD monitor, PDA).

Je dobře známým, ale již méně respektovaným faktem, že design dokumentu musí být znovuvytvořen pro každé nové výstupní zařízení. Mnoho dokumentů vyladěných TEXovým systémem pro čtení z výtisku na papíře (mikrotypografie atd.) je hrdě vystaveno na webu bez přizpůsobení designu pro konkrétní výstupní zařízení (LCD obrazovka, PDA), pro konkrétní účel a pro specifické požadavky čtenáře. To je v rozporu s cílem, který měl na mysli Knuth např. při návrhu fontů v METAFONTu – dokonce i maličké detaily rasterizace ovlivněné různými tiskárnami by měly být vyladěny správným nastavením pro konkrétní tiskárnu v souboru parametrů `modes.mf`.

Autoři, kteří užívají systém s otevřeným zdrojovým kódem na bázi TEXu, mají značné možnosti k ovlivnění každého aspektu formy při psaní svých článků a knih. Pokud autoři při psaní používají *pouze* logické značkování, je možné nezávisle

Děkujeme za podporu České akademie věd granty AV1ET208050401 a AV1ET200190513, stejně jako za cestovní příspěvek Československého sdružení uživatelů TEXu.

zvolit odlišnou typografickou úpravu – respektující obsah a odpovídající různým možnostem výstupních zařízení – změnou vizuální reprezentace jednotlivých logických částí textu. Důsledné oddělení obsahu a formy je možné téměř vždy, pouze s ojedinělými výjimkami, jako je sazba básní Christiana Morgensterna. Je však nutná disciplinovanost autorů při vyhýbání se přímého užití vizuálních příkazů jako je např. `\vskip`: důsledná lokalizace vizuálního formátování do stylu sazby je téměř nezbytností.

Jak roste kvalita a rozlišení zobrazovacích zařízení, stále delší texty jsou čteny přímo z obrazovky počítače či z mobilních telefonů nebo PDA. Autoři přirozeně vyžadují, aby obsah jejich dokumentů *byl* přizpůsoben pro širokou řadu různých výstupních zařízení. I když návrh designu dokumentu je podstatně obtížnější pokud musí být vzata v úvahu celá řada těchto výstupních zařízení a prohlížečů, je to přesně to, co je požadováno čtenáři.

V tomto článku zmíníme několik publikačních projektů: dvě matematická skripta [1, 2] a databázové publikování v DML-CZ [11]. Ve všech projektech jsou ukázány výhody plynoucí ze striktního oddělení formy a obsahu pro užití jednoho vstupního zdroje vytvořeného autorem nebo vygenerovaného na požádání z databáze pro různé typy výstupů.

Publikování z jednoho zdroje

If the only tool we know is a word processor, everything looks like a print document.
–Peter Meyer [9]

Autor chce předávat informace. A jediná věc, kterou *musí* udělat, je označit logické prvky v textu. To umožní designerovi vynutit konzistenci: vizuální zobrazení stejných logických částí by mělo být důsledně stejné. Vyznačení logických prvků v textu jeho autorem umožní publikování pro různá výstupní zařízení pouhým přepnutím mezi různými designy. Údržba textu je pak mnohem jednodušší a levnější než údržba různých verzí obsahově stejného textu, které se liší pouze zamýšleným způsobem použití. Také to snižuje počet chyb, zlepšuje konzistenci a/nebo šetří náklady na překlad. Pro tento typ pořizování a zpracování dokumentů se užívá termín *publikování z jednoho zdroje* (*single-source publishing* nebo *single-sourcing*) – `http://en.wikipedia.org/wiki/Single_source_publishing`. Dobře známým systémem a sadou DTD umožňujících publikování z jednoho zdroje je DocBook, který obsahuje podporu pro konverzi v něm označkovaných dokumentů do XHTML, DVI, PS nebo PDF. A to buď za pomoci XSL, XSL-FO, nebo LaTeXem.

Publikování z jednoho zdroje často vychází z XML jako zdroje obsahu [9]. Pro mnoho autorů však není dost pohodlné zapisovat XML přímo, a to ani s užitím „chytrého“ XML editoru. Technické rukopisy plné matematiky budou i nadále pořizovány v nějaké formě TeXu díky své kompaktnosti, přehlednosti

a čistotě matematického zápisu TEXu. Produktivita autora znatelně vzrůstá s „autor-centrickými“ systémy [8], v případě matematiky např. s $\mathcal{AMS}$-LATEXem.

V akademické oblasti autoři připravují učební materiály pro své předměty sami a chtějí je publikovat ve formátech, které upřednostňují jejich studenti. Metodou publikování z jednoho zdroje jsme proto připravili do různých výstupních formátů dvě matematická skripta [1, 2] ze zdrojového textu autorsky pořízeného v LATEXu. V dalších částech popíšeme naše zkušenosti s těmito projekty.

Značkování a konverzní nástroje

Data cannot be used at a finer grain than it is marked up at.
–Rick Jelliffe

Abychom umožnili publikování z jednoho zdroje, museli jsme podstatně pročistit zdrojové soubory, protože ty původně vůbec nebyly psány s cílem publikování do více odlišných formátů. I Donald Ervin Knuth, DEK, psal velmi „nízkoúrovňový“ kód ve svém TEXbooku:

```
&\elevenit I\kern.7ptllustrations by\cr
&DU\kern-1ptANE BIBBY\cr
\noalign{\vfill}
&\setbox0=\hbox{\manual77}%
\setbox2=\hbox to\wd0{\hss\manual6\hss}%
\raise2.3mm\box2\kern-\wd0\box0\cr % A-W logo
&ADDISON\kern.1em--WESLEY\cr
%&PUBLISHING COMP\kern-.13emANY\kern-1.5mm\cr
```

Neočekával totiž, že by kód sloužil pro cokoli jiného než pro přípravu sazby pro tisk na fotosázecím stroji – s danými fonty, kerningem pro danou velikost atd. Pro potřeby publikování z jednoho zdroje musí být hlavní text napsán bez ladění pro konkrétní výstupní zařízení/tiskárnu a nízkoúrovňové značkování musí být nahrazeno vysokoúrovňovým značkováním, které umožňuje více odlišných definic použitých maker pro různé výstupy. Značkování musí být zvoleno s nejlepší možnou granularitou, převoditelné na odpovídající nastavení designu pro každý typ výstupu. Není to nová myšlenka a je užita také ve světě XML (rozdílné nastavení zobrazení pomocí kaskádových stylů CSS pro odlišná zařízení nebo prohlížeče).

Určili jsme několik typů výstupních formátů, které naši studenti požadovali. Vedle standardní verze vhodné pro tisk na papír byla požadována prohledávatelná verze dokumentu optimalizovaná pro LCD obrazovku s poměrem stran 4:3. Pro některé účely byla potřebná i (X)HTML verze dokumentů pro platformy a zařízení bez PDF prohlížečů. Nakonec jsme připravili také XHTML+MathML verzi skript, což je vhodný formát např. pro nevidomé.

Existuje mnoho nástrojů a pomůcek pro konverzi TEXových dokumentů do různých výstupních formátů – seznam některých z nich je dostupný na webové stránce TUGu *TeX Resources on the Web* (`http://www.tug.org/interest.html`). PDFLATEX s balíčky `hyperref` a `crop` je vhodná kombinace pro tiskový výstup. Pro verzi dokumentu pro prohlížení na obrazovce jsme zvolili balíček `pdfscreen` v kombinaci s PDFTEXem a balíčkem `hyperref`.

PDF verze

Make all visual distinctions as subtle as possible, but still clear and effective.
–Edward R. Tufte [17]

Pro každou verzi výstupu byly parametry designu a maker zapsány v oddělené podmíněné větvi. Ukázka jednoduchého zdrojového kódu:

```
\newif\ifprint
\printfalse % Obrazovková verze.
%\printtrue % Tisková verze.

\ifprint
    \hypersetup{colorlinks=false, pdfborder={0 0 0}}
    % Centrovaný zrcadlený dokument na papíru
    % formátu A4 s přidáním ořezových značek.
    \usepackage[cam,a4,center,mirror]{crop}
\fi
```

PDF pro tisk

Pro potřeby tisku je vhodné připravit výstup v odstínech šedi. K tomuto účelu je často užíván balíček `hyperref` s odpovídajícím nastavením (`colorlinks=false`, `pdfborder={0 0 0}`).

Balíček `crop` (dostupný na CTAN) je dobrá volba pro přidání ořezových značek do dokumentu. Balíček `crop` je také schopen provést některé transformace dokumentu, jako je jeho zrcadlení apod. Výstup je ukázán na obrázku 1 na straně 123.

Výhodou balíčku `crop` je to, že tyto transformace je schopen s dokumentem provádět dodatečně, bez redefinice geometrie dokumentu ze strany uživatele. Je snadné vzít již existující, finálně zalomený dokument, a přidáním jediného řádku vytvořit jeho tiskovou variantu bez nutnosti hlubších zásahů, a aniž by bylo nutné cokoli znovu ladit nebo by hrozilo nebezpečí změny zlomu dokumentu.

PDF pro prohlížení na obrazovce

Typický LCD monitor je úplně odlišné výstupní zařízení než tiskový výstup. V porovnání s dnešními domácími tiskárnami s rozlišením 1 200 a více DPI

má obrazovka řádově nižší rozlišení. U elektronické publikace můžeme využít interaktivních možností, které nám tento formát nabízí – v dokumentu je možné používat hypertextové odkazy, barvy, navigační lišty atd.

Pro obrazovkovou verzi učebnic jsme použili balíček `pdfscreen` (dostupný na CTAN). Kód pro `pdfscreen` byl spolu s definicemi prostředí a maker pro obrazovku opět v oddělené podmíněné větvi stylopisu.

Abychom se vyhnuli náročnému opakovanému ladění řádkového zlomu, definovali jsme geometrii tiskového zrcadla obrazovkové verze dokumentu tak, aby byl řádkový zlom identický jako v tiskové verzi. Vzhledem k značně odlišné geometrii stránky je však odlišný stránkový zlom, jak je ukázáno na obrázku 2 na straně 124. Zachování stejného řádkového zlomu znamenalo velkou úsporu času a námahy sazeče, bohužel neexistuje způsob, jak toto provést automaticky. Nastavení odpovídající geometrie tiskového zrcadla tedy stejně vyžadovalo jisté úsilí.

Webové verze dokumentu

Hitem dneška je XML – a jednou z oblastí využití XML technologií je dnešní web, který je také místem, kam ve formátu (X)HTML míří značná část dnes produkovaných dokumentů. Pro vytváření prohledávatelných a škálovatelných matematických zápisů pro web je pak vhodným standardem XML jazyk MathML.

Pro převod TEXových dokumentů do webových formátů je k dispozici několik nástrojů. Patří mezi ně např.:

- TEX2page (`http://www.ccs.neu.edu/home/dorai/tex2page/`),
- Tralics (`http://www-sop.inria.fr/apics/tralics/`),
- TEX4ht (`http://www.cse.ohio-state.edu/~gurari/TeX4ht/`),
- LATEXML (`http://dlmf.nist.gov/LaTeXML/`) nebo
- LATEX2HTML (`http://www.latex2html.org/`).

Každý nástroj má některé výhody, ale i nevýhody; my zde nechceme rozebírat všechny z nich. Po otestování některých z těchto nástrojů jsme pro přípravu skript vybrali program TEX4ht [3, 6], který nabízí převod jak do HTML, tak do XHTML. Navíc podporuje i převod matematiky do značkování MathML.

Značnou výhodou TEX4ht je, že využívá nativní překladač TEXu pro přípravu standardních DVI výstupních souborů, které však obsahují `\special` příkazy pro následné zpracování. Díky tomu nehrozí nebezpečí „nepochopení“ nepodporovaných příkazů jako např. v případě LATEX2HTML. Teprve z takto obohaceného DVI jsou následně vygenerovány (X)HTML stránky dalšími nástroji TEX4ht.

HTML

Nejobtížnější částí konverze byla příprava webového výstupu. V případě komplexních dokumentů je obvykle nutné provést některé změny ve zdrojovém kódu. Již existující logické značkování LaTeXu bylo využito kdekoli to bylo možné. Tyto modifikace a příkazy specifické pro TeX4ht byly opět v odděleném stylovém souboru.

Základní použití TeX4ht je velmi jednoduché. Když máme zdrojový kód ve formátu LaTeX s načteným TeX4ht stylem, můžeme vyzkoušet provést konverzi dokumentu. Nejdříve můžeme vyzkoušet příkaz:

```
htlatex dokument.tex 'html'
```

Pokud TeX4ht úspěšně dokončí svou práci, výsledkem je HTML podoba našeho původního dokumentu. S tímto nastavením jsou komplikovanější matematické vzorce vykresleny do PNG obrázků, jak můžete vidět na obrázku 3 na straně 125. Tato cesta je oproti MathML zápisu jednoduchá a bezpečná pro čtení všemi webovými prohlížeči, včetně jejich starších verzí.

Pokud není specifikováno jinak, tak je celý dokument převeden do jediného HTML souboru. TeX4ht je však schopen dlouhý dokument automaticky rozdělit do stromové struktury vzájemně propojených webových stránek. Volání překladu příkazem `htlatex dokument.tex 'html,2'` vygeneruje dokument s každou kapitolou v odděleném souboru. Navigace mezi kapitolami je pak možná pomocí lišty s odkazy, která je přidána na horní a spodní kraj každé stránky.

Při generování HTML výstupu může být užitečné mít možnost do dokumentu přímo vložit nějaký HTML kód. To provedeme příkazem `\HCode{Nějaký HTML kód.}`. Ten může být použit také pro vložení CSS kódu.

Pro definici CSS jsme užili příkaz `\Css{Definice CSS}`. CSS atributy jsou mapovány na patřičné elementy dokumentu pomocí kódu vkládaného do výstupu pomocí obecného příkazu `\HCode`. Příkaz `\ConfigureEnv` pak zajišťuje pohodlnou definici kódu vkládaného před a za obsah odpovídajícího LaTeXového okolí, a to bez nutnosti zásahu přímo do nativní definice okolí.

Jako trošičku složitější příklad definice CSS může posloužit prostředí `veta`:

```
\newtheorem{veta}{Věta}[chapter]
...
\ifweb % Při vytváření webového výstupu...
  \Css{% Definice CSS kódu
      .veta { background-color: \#FFFFFF;
              border: 1px solid;
              border-color: \#0000FF; } }
  % Ve značkování výsledného HTML dokumentu je před každé prostředí 'veta'
  % umístěna značka '<div class="veta">', za prostředí je umístěna
  % značka '</div>'.
  \ConfigureEnv{veta}
      {\HCode{<div class="veta">}}
```

```
      {\HCode{</div>}}
      {}{}
\fi
...
% V dokumentu je užito stejné LaTeXové značkování pro všechny verze
% prostředí 'veta'.
\begin{veta}
    Funkce~$f(x,y)$ je spojitá v~bodě~$[x_{0},y_{0}]$.
\end{veta}
```

XHTML + MathML

TEX je velmi často používán pro sazbu vědeckých textů, ve kterých je užito velké množství matematických zápisů. Pokud mají být výstupní XHTML dokumenty maximálně využitelné, tak mnohem zajímavější alternativou k HTML s matematikou v obrázcích je rozšíření XHTML o značkování XML jazyka MathML. Nejkomplikovanější konverzní proces, který jsme na matematických skriptech vyzkoušeli, byla právě konverze do XHTML + MathML.

Bohužel, při užití MathML jsou zde v současnosti jisté obtíže nejen pro autory, ale i uživatele. Zaprvé, implementace MathML ve webových prohlížečích je roztříštěná. Nejlepších výsledků jsme s TEX4ht dosáhli při použití volby překladu `mozilla` v kombinaci s webovým prohlížečem Mozilla Firefox (nebo jiným prohlížečem založeným na jádře Gecko) použitým pro prohlížení výsledného dokumentu. Zadruhé, uživatelé musí mít na svém počítači nainstalovány odpovídající matematické fonty. Pro uživatele prohlížeče Mozilla Firefox jsou informace o potřebných fontech, odkazy k jejich stažení a instalační pokyny dostupné z webové stránky *MathML in Mozilla* (`http://www.mozilla.org/projects/mathml/fonts/`).

Velkou výhodou TEX4ht je mimo jiné také možnost MathML výstupu, který je velmi užitečný v případě matematických textů – generování XHTML + MathML je přitom velmi obdobné generování HTML:

```
htlatex dokument.tex 'xhtml,mozilla'
```

Pokud vše proběhne, jak má, obdržíme po skončení překladu XML soubor obsahující XHTML kód, který používá jazyk MathML pro vyjádření matematických zápisů. Výsledek, plně škálovatelný dokument, můžete vidět na snímku okna prohlížeče Mozilla Firefox na obrázku 4 na straně 126.

Komplikací pro autory je mimo jiné to, že TEX4ht je při generování MathML velmi citlivý na čistotu zápisu matematiky ve zdrojovém textu dokumentu. Např. zápis

```
$M=\{x|x$ je liché $\}$
```

je korektní TEXový zápis. TEX4ht ale v tomto případě nemá informaci o párování složených závorek. Je tedy nutné použít správnější zápis:

```
$M=\{x|x \mbox{je liché}\}$
```

Toto byla značná komplikace při převodu matematických skript, která původně nebyla psána s ohledem na tuto potřebu překladače.

Pro $n = 2$ budeme místo $f(x_1, x_2)$ psát $f(x, y)$ a pro $n = 3$ místo $f(x_1, x_2, x_3)$ píšeme $f(x, y, z)$.

Příklad 1.1. i) Zobrazte v rovině definiční obor funkce

$$f(x, y) = \sqrt{\left(x^2 + \frac{(y-2)^2}{4} - 1\right)\left(x^2 + y^2 - 6x\right)}.$$

Řešení. Výraz pod odmocninou musí být nezáporný, tj. musí být splněna podmínka

$$\left(\frac{(y-2)^2}{4} + x^2 - 1\right)\left(x^2 + y^2 - 6x\right) \geq 0.$$

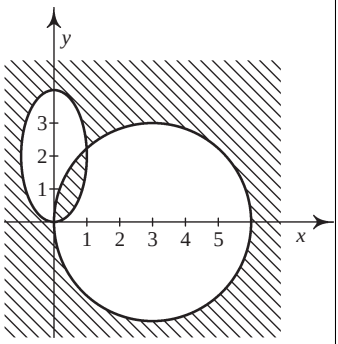
To nastane, právě když

$$\frac{(y-2)^2}{4} + x^2 - 1 \geq 0 \quad \text{a} \quad (x^2 + y^2 - 6x) \geq 0$$

nebo

$$\frac{(y-2)^2}{4} + x^2 - 1 \leq 0 \quad \text{a} \quad (x^2 + y^2 - 6x) \leq 0.$$

Rovnice $\frac{(y-2)^2}{4} + x^2 = 1$ je rovnicí elipsy se středem v bodě $[0, 2]$ a poloosami délek $a = 1$ a $b = 2$, rovnice $x^2 + y^2 - 6x = 0$ je rovnicí kružnice se středem v bodě $[3, 0]$ a poloměrem $r = 3$, neboť tuto rovnici lze převést na tvar $(x-3)^2 + y^2 = 9$. Množina všech bodů $[x, y] \in \mathbb{R}^2$ splňující výše uvedené nerovnosti, tj. definiční obor funkce f, je znázorněna na vedlejším obrázku. Je to uzavřená množina v $\mathbb{R}^2$.



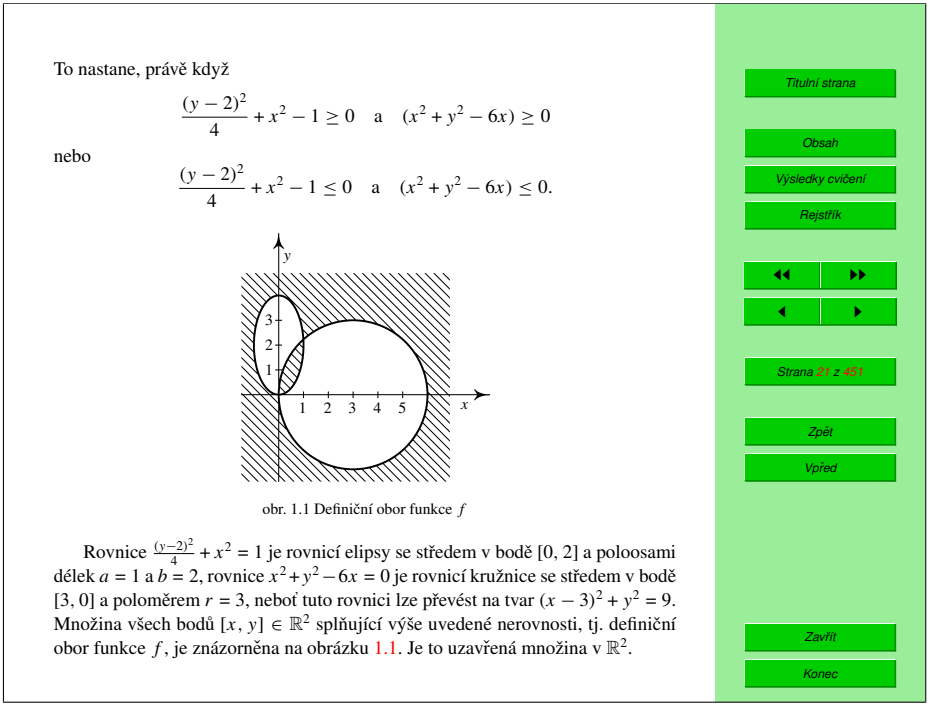
ii) Zobrazte v rovině definiční obor funkce

$$f(x, y) = \arccos(x^2 + y^2 - 1) + \sqrt{|x| + |y| - \sqrt{2}}.$$

Řešení. Definičním oborem funkce arccos je interval $[-1, 1]$, první sčítanec je tedy definován pro $[x, y]$ splňující nerovnosti

$$-1 \leq x^2 + y^2 - 1 \leq 1,$$

Obrázek 1: Tiskový výstup (bez zrcadlení stránek)

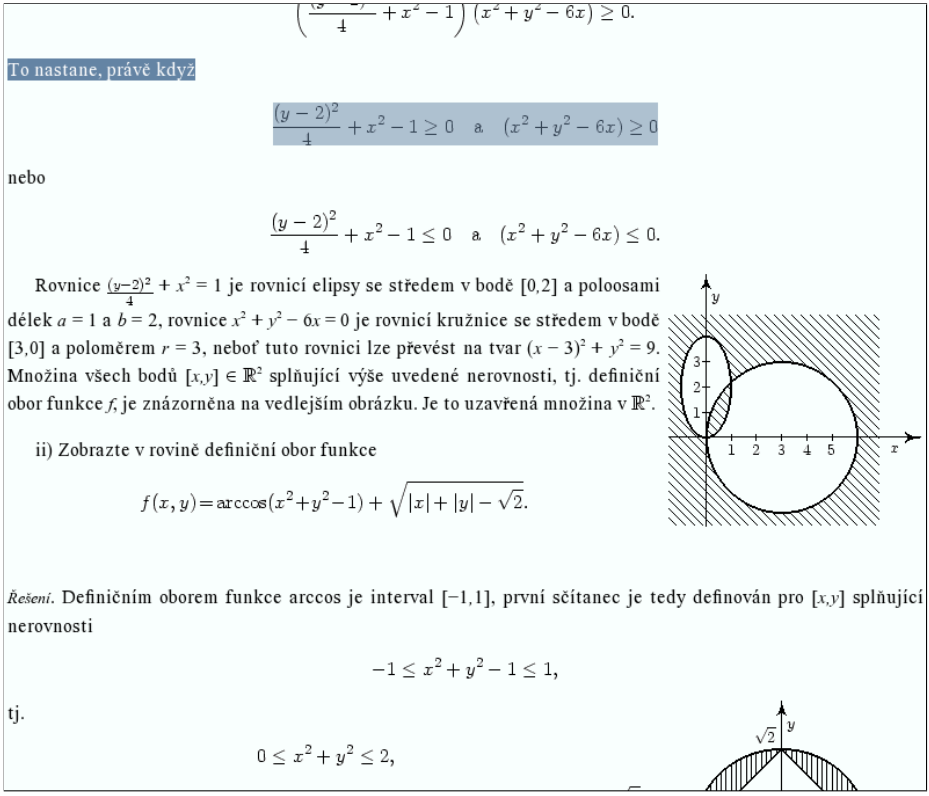


Obrázek 2: Obrazovkový výstup

Ve složitých případech může být přepis matematického zápisu velmi komplikovaný, nebo dokonce nemožný. Pro tyto vzácné, ale přesto hrozící případy TEX4ht nabízí nouzové řešení – uživatel si může u určitého úseku kódu vynutit jeho obrázkovou reprezentaci ve výstupním dokumentu.

```
\ifweb
    \Picture*{}
\fi
$M=\{x|x$ je liché $\}$
\ifweb
    \EndPicture
\fi
```

Toto řešení není omezeno pouze na zápis matematiky, ale může být užito pro libovolný objekt s problematickou reprezentací v (X)HTML/MathML.



$$\left(\frac{(y-2)^2}{4}+x^2-1\right)(x^2+y^2-6x)\geq 0.$$

To nastane, právě když

$$\frac{(y-2)^2}{4}+x^2-1\geq 0 \quad \text{a} \quad (x^2+y^2-6x)\geq 0$$

nebo

$$\frac{(y-2)^2}{4}+x^2-1\leq 0 \quad \text{a} \quad (x^2+y^2-6x)\leq 0.$$

Rovnice $\frac{(y-2)^2}{4}+x^2=1$ je rovnicí elipsy se středem v bodě $[0,2]$ a poloosami délek $a=1$ a $b=2$, rovnice $x^2+y^2-6x=0$ je rovnicí kružnice se středem v bodě $[3,0]$ a poloměrem $r=3$, neboť tuto rovnici lze převést na tvar $(x-3)^2+y^2=9$. Množina všech bodů $[x,y]\in\mathbb{R}^2$ splňující výše uvedené nerovnosti, tj. definiční obor funkce f, je znázorněna na vedlejším obrázku. Je to uzavřená množina v $\mathbb{R}^2$.

ii) Zobrazte v rovině definiční obor funkce

$$f(x,y)=\arccos(x^2+y^2-1)+\sqrt{|x|+|y|-\sqrt{2}}.$$

Řešení. Definičním oborem funkce arccos je interval $[-1,1]$, první sčítanec je tedy definován pro $[x,y]$ splňující nerovnosti

$$-1\leq x^2+y^2-1\leq 1,$$

tj.

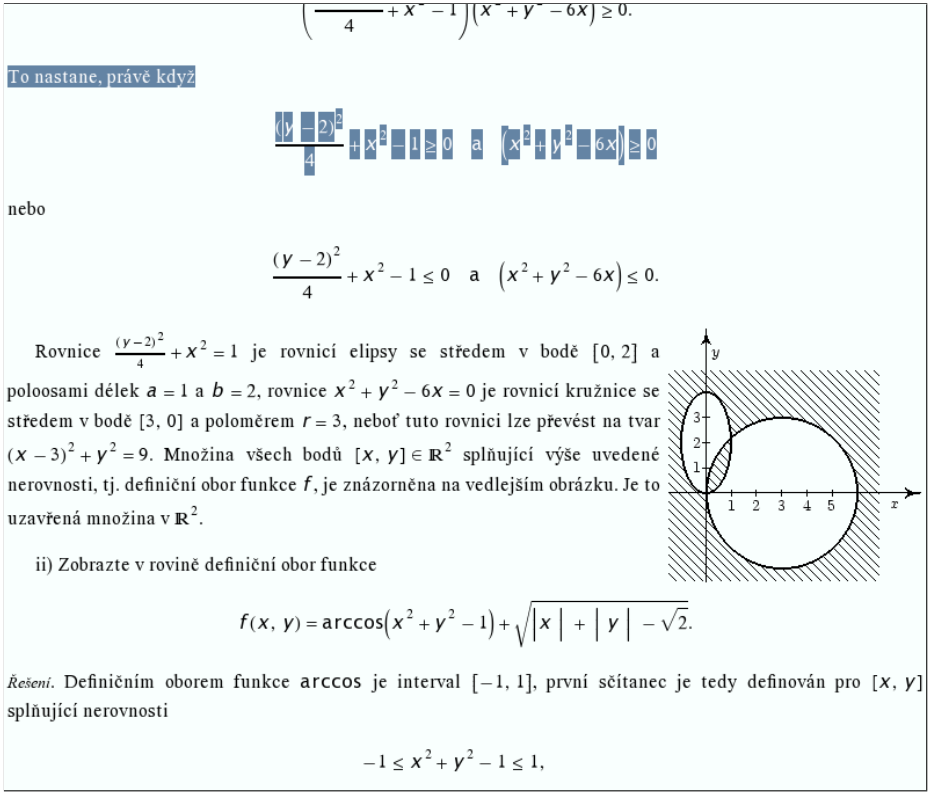
$$0\leq x^2+y^2\leq 2,$$

Obrázek 3: HTML výstup

Publikování digitalizovaných textů

Druhý projekt, kde je využívána cesta publikování z jednoho primárního zdroje, je projekt DML-CZ `http://dml.cz` [11, 15]. Cílem projektu není jen digitalizace 250 000 stran českých a slovenských matematických časopisů, ale také poskytnutí nástrojů pro souběžné generování tiskových a pro web optimalizovaných verzí nových, digitálně pořizovaných čísel časopisů.

Zpracování začíná naskenováním převážně bitonálních TIFF obrázků stránek dokumentů v rozlišení 600 DPI. Následný OCR proces je prováděn v několika fázích za užití programů FineReader (OCR textu) a InftyReader (OCR matematiky) [12, 14]. FineReader pro výstup OCR umí využít uložení více vrstev do PDF, a pod naskenovaný obrázek stránky umí uložit rozpoznaný text (pro vyhledávání, indexování). InftyReader [16] umí takové PDF načíst a je scho-



Obrázek 4: MathML výstup

pen exportovat text včetně matematických strukturních vzorců v LaTeXu nebo v MathML spolu s pozičními informacemi o umístění objejktů. To případně umožní vysázet rozpoznaná LaTeXová a/nebo MathML data jako další vrstvy PDF pod obrázek, například PDFLaTeXem za použití standardního prostředí `picture`. Pro vytvoření prohledávatelných dokumentů v různých jazycích je nutné použít balíček `cmap`.

Velikost vygenerovaných souborů je důležitá – pro tisk je obvykle požadováno rozlišení tiskového zařízení, pro prohlížení na obrazovce je dostačující nižší rozlišení. PDF nabízí odlišné typy kompresních filtrů – od verze 1.4 PDF standardu je podporována komprese bitonálních obrázků metodou JBIG2. PDFTeX ve svých posledních verzích JBIG2 komprimované obrázky podporuje také. Navíc Adam Langley napsal open-source konverzní program `jbig2enc` jako projekt podporovaný společností Google [7].

JBIG2 nabízí jak bezztrátovou, tak ztrátovou kompresi. Pro naskenovaný obsah je nejvhodnějším postupem k získání nejlepšího kompresního poměru přijmutí drobných chyb v obrazových datech užitím symbolického kódování, kde variace vznikají z tiskových chyb. Kompresní poměr JBIG2 závisí na velikosti kontextu – lepší výsledky jsou dosahovány, pokud se komprimují celé rozsahy stran místo jediné oddělené stránky. Při užití lehce ztrátové komprese a velké velikosti kontextu jsme schopni generovat PDF obrázky, které mají zhruba asi jen 10–20% velikost oproti CCITT kódovaným a LZW komprimovaným obrázkům. Parametry programu `jbig2enc` umožňují finální vyladění regenerace obrázků, a to i pro jejich „on-the-fly“ generování [5], které bylo implementováno pro OSS systém Kramerius.

Nové články časopis˚ u digitální knihovny se ji˘ z samozřejmě naskenují, ale přebírají sázené (TEXem). Pro sazbu časopisu *Archivum Mathematicum* [10] se zvažuje i generování nových článků v XHTML+MathML spolu s tiskovou verzí v PDF.

Jelikož se jak formát PDF, tak požadavky a možnosti čtenářů mění, základním principem je uchovávat (a editovat) pouze *jedna primární data* a metadata, a všechny aktuálně vystavované výstupy z nich generovat automatizovanými procedurami a programy. Tak jsou například před závěrečným importem do digitální knihovny DSpace stránky článku spojeny do jednoho PDF, přigenerována úvodní stránka s metadaty a citačními informacemi, PDF je optimalizováno do aktuálně vhodné verze PDF s již široce akceptovanými kompresními filtry a výsledné PDF určené k šíření je digitálně podepsáno prostředy PKI infrastruktury. To vše se děje skripty, které se dají spouštět opakovaně po rozsáhlejších úpravách primárních (meta)dat, případně se dají vytvářet varianty generujících skriptů pro různá užití výstupů (tagged PDF, PDF optimalizovaná pro web ap.).

Závěr

Na příkladech několika projektů jsme ukázali, že na TEXu založené pořizování autorského textu a publikování z jednoho zdroje či jedněch primárních dat je velmi přirozená a efektivní cesta přípravy a personalizace dokumentů pro různá výstupní zařízení. TEX4ht je velmi konfigurovatelný nástroj pro webové publikování z TEXových zdrojů, JBIG2 formát pro bitonální obrázky značně šetří diskové a přenosové kapacity bez újmy na kvalitě, pokud se generuje velké množství obrázků jako tomu je v případě digitalizačních projektů jako DML-CZ.

Reference

[1] Zuzana Došlá and Ondřej Došlý. *Metric Spaces: Theory and Examples (in Czech)*. Masaryk University in Brno, second edition, 2000.

[2] Zuzana Došlá, Roman Plch, and Petr Sojka. Matematická analýza s programem Maple: 1. Diferenciální počet funkcí více proměnných (Mathematical Analysis with Program Maple: 1. Differential Calculus). CD-ROM, http://www.math.muni.cz/~plch/mapm/, December 1999.

[3] Michel Goossens, Sebastian Rahtz, Ross Moore, and Bob Sutor. *The LaTeX Web Companion*. Addison-Wesley, Reading, MA, 1999.

[4] Philip Babcock Gove and Merriam Webster. *Webster's Third New International Dictionary of the English language Unabridged*. Merriam-Webster Inc., Springfield, Massachusetts, U.S.A, January 2002.

[5] Miroslav Grabovský. Generování PDF pro systém Kramerius. Master's thesis, Masaryk University, Brno, Faculty of Informatics, January 2007.

[6] Eitan M. Gurari. TeX4ht: LaTeX and TeX for Hypertext. http://www.cse.ohio-state.edu/~gurari/TeX4ht/, February 2005.

[7] Adam Langley and Dan S. Bloomberg. Google Books: Making the public domain universally accessible. In *Proceedings of SPIE — Volume 6500, Document Recognition and Retrieval XIV*, pages 1–10, San Jose, CA, January 2007. The International Society of Optical Engineering. http://www.imperialviolet.org/binary/google-books-pdf.pdf.

[8] Peter Meyer. Planning a single source publishing application for business documents, 2005. http://www.elkera.com/cms/articles/seminars_and_presentations/.

[9] Peter Meyer. Introduction to single source publishing, 2006. http://www.elkera.com/cms/articles/technical_papers/.

[10] Michal Růžička. Automated Processing of TeX-typeset Articles for a Digital Library. In Sojka [13], pages 167–176.

[11] Petr Sojka. From Scanned Image to Knowledge Sharing. In Klaus Tochtermann and Hermann Maurer, editors, *Proceedings of I-KNOW '05: Fifth International Conference on Knowledge Management*, pages 664–672, Graz, Austria, June 2005. Know-Center in coop. with Graz Uni, Joanneum Research and Springer Pub. Co.

[12] Petr Sojka. Towards Digital Mathematical Library: Optical Character Recognition on Mathematical Texts. In Julius Štuller and Zdenka Linková, editors, *Inteligentní modely, algoritmy a nástroje pro vytváření sémantického webu*, pages 110–113, Praha, Czech Republic, 2006. Ústav informatiky AV ČR.

[13] Petr Sojka, editor. *Towards Digital Mathematics Library—Proceedings of DML 2008*, Birmingham, UK, July 2008. Masaryk University.

[14] Petr Sojka, Radovan Panák, and Tomáš Mudrák. Optical Character Recognition of Mathematical Texts in the DML-CZ Project. Technical report, Masaryk University, Brno, September 2006. presented at CMDE 2006 conference in Aveiro, Portugal.

[15] Petr Sojka and Jiří Rákosník. From Pixels and Minds to the Mathematical

Knowledge in a Digital Library. In Sojka [13], pages 17–27.
[16] Masakazu Suzuki, Fumikazu Tamari, Ryoji Fukuda, Seiichi Uchida, and Toshihiro Kanahori. INFTY — An integrated OCR system for mathematical documents. In C. Vanoirbeek, C. Roisin, and E. Munson, editors, *Proceedings of ACM Symposium on Document Engineering 2003*, pages 95–104, Grenoble, France, 2003. ACM.
[17] Edward R. Tufte. *Visual Explanations.* Graphics Press LLC, 1997.

Summary: Parallel Electronic Publications

TEX is traditionally used as an authoring tool for the paper publishing of scientific texts and textbooks. Parallel electronic publications that are meant for on-screen viewing and web delivery are also demanded by readers for many reasons today. This paper discusses the ways to single-source author publishing from a LATEX source file, and it shows examples of several textbooks published by this approach. Special attention is given to the web document generation either to HTML or XHTML markup with a notation translated to MathML. Also discussed is a personalised automated document generation for a digital library project DML-CZ.

Fakulta informatiky Masarykovy univerzity
Botanická 68a, 602 00 Brno, Česká republika
http://www.fi.muni.cz/usr/sojka/
sojka@fi.muni.cz, xruzick7@fi.muni.cz