

# Zpravodaj Československého sdružení uživatelů TeXu

---

Vít Zýka

Používáme pdfTeX: vkládání obrázků

*Zpravodaj Československého sdružení uživatelů TeXu*, Vol. 11 (2001), No. 4, 181–186

Persistent URL: <http://dml.cz/dmlcz/150302>

## Terms of use:

© Československé sdružení uživatelů TeXu, 2001

Institute of Mathematics of the Czech Academy of Sciences provides access to digitized documents strictly for personal use. Each copy of any part of this document must contain these *Terms of use*.



This document has been digitized, optimized for electronic delivery and stamped with digital signature within the project *DML-CZ*:  
*The Czech Digital Mathematics Library* <http://dml.cz>

*Program pdf $\TeX$  sdružuje vlastnosti samotného  $\TeX$ u nezávislé na výstupním zařízení, ale i ty hardwarově závislé, které  $\TeX$  přenechává DVI ovladačům. Příkladem takového tiskového materiálu jsou obrázky. Současný pdf $\TeX$  podporuje čtyři grafické formáty (PDF, PNG, JPEG a TIFF) a pátý (METAPOST) díky makrokonverzi do PDF. Tento článek popisuje jejich vkládání na nejnižší úrovni pdf $\TeX$  primitiv a ukazuje způsob, jak obrázek opakovat, třeba i v jiné velikosti, aniž by byl v dokumentu fyzicky vícekrát přítomen.*

## Podporované grafické formáty a jejich vhodné použití

**PDF** (Portable Document Format<sup>1</sup>) Přirozený a jediný podporovaný vektorový formát podporující ztrátovou i bezztrátovou kompresi. **Způsob načtení při překladu:** Soubor se vloží do dokumentu beze změny stupně komprese. Načtení obrázku při překladu pdf $\TeX$ em je tak velmi rychlé. **Doporučené použití:** Vektorová data, případně i bitmapy, zvláště ty, které nebudeme skladovat ve formátu PNG, JPG či TIFF nebo jde-li nám o překladový čas, třeba v době ladění dokumentu.

**PNG** (Portable Network Graphics<sup>2</sup>, World Wide Web Consortium – W3C) Bitmapový formát s možností bezztrátové komprese. Umí zaznamenat všechny nejčastěji používané barevné prostory (od 2 do 16 bitů na složku). Ve verzi pdf $\TeX$ u 14h však vznikají problémy s barevným prostorem CMYK a  $\alpha$ -kanálem. RGB nebo šedotónový prostor jsou bez problémů. **Způsob načtení:** Před vložením se soubor nejprve dekomprimuje, aby se opět zkomprimoval podle aktuálně nastavené hodnoty `\pdfcompresslevel`. Načítání je pomalejší. **Použití:** Černobílé bitmapy a materiál technické povahy (scanované výkresy, mapy) vyžadující bezztrátovou kompresi pro zachování hranové ostroty.

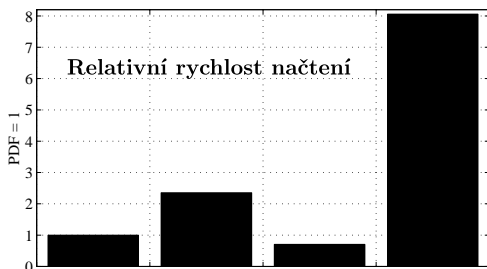
**JPEG** (Joint Photographic Experts Group<sup>3</sup>) Bitmapový formát se ztrátovou kompresí. **Způsob načtení:** Soubor je do dokumentu vložen bez dalších úprav. Načtení je proto nejrychlejší. **Použití:** Šedotónové a barevné fotografie.

---

<sup>1</sup>Specifikace na <http://partners.adobe.com/asn/developer/acrosdk/DOCS/PDFRef.pdf>

<sup>2</sup><http://www.w3.org/TR/REC-png>

<sup>3</sup><http://www.jpeg.org>



(a) PDF



(b) Metapost



(c) PNG



(d) JPG

Obrázek 1: Příklad vhodných grafických formátů pro různé typy obrázků. V grafu je znázorněna relativní rychlost načtení podporovaných formátů.

**TIFF** (Tagged Image File Format<sup>4</sup>) Velmi obecný formát, ovšem pdfTeX podporuje jen část jeho specifikace. Z tohoto důvodu doporučuji raději používat funkčně podobný formát PNG. PNG je také obvykle o něco úspornější. **Způsob načtení:** Dekomprese a opětovná komprese s nejpomalejším zpracováním. **Použití:** Podobné jako PNG.

**METAPOST** [1] Ačkoliv výstupem METAPOSTu je Encapsulated PostScript, který není pdfTeXem podporován, lze jej použít. EPS generované METAPOSTem je tak jednoduché, že Hans Hagen napsal T<sub>E</sub>Xová makra, která umožňují přímou konverzi do PDF. Stačí načíst balík<sup>5</sup>

```
\input supp-pdf
```

a použít makro

```
\convertMPtoPDF{kvitek.1}{0.8}{0.8}
```

Prvním parametrem je název souboru, zbylé dva koeficienty znamenají horizontální respektive vertikální zvětšení.

<sup>4</sup><http://partners.adobe.com/asn/developer/pdfs/tn/TIFF6.pdf>

<sup>5</sup>Jde o sadu maker z formátu Context, ale lze ji použít i pro L<sup>A</sup>T<sub>E</sub>X a plain T<sub>E</sub>X.

## Vkládání obrázků

Pro vkládání obrázků na nejnižší úrovni slouží tři primitiva pdf<sub>T</sub>E<sub>X</sub>u [3] s následujícím použitím:

```
\immediate\pdfximage width 2cm height 1cm depth 0pt{file.ext}
\hbox{\pdfrefximage\pdflastximage}
```

První příkaz vloží do dokumentu externí objekt (v jazyku PDF /Type /XObject /Subtype /Image) a jeho číslo vloží do registru `\pdflastximage`. Formát souboru rozpozná podle jeho koncovky (pdf, png, jpg i jpeg a tif i tiff). Uvedení rozměrů typu *⟨rule specification⟩* před jménem souboru nerozhoduje o jeho uložení (obrázek je vždy vložen v přirozené velikosti), ale použije se až při rendrování stránky. Neuvede-li se hloubka (depth) předpokládá se nulová. Je-li uveden pouze horizontální (width) nebo pouze vertikální (height+depth) rozměr, pak se druhý dopočítá při zachování poměru stran. Při obou udaných rozměrech se obraz deformuje. Není-li uveden žádný rozměr, použije se buď hodnota v grafickém formátu a není-li ani tam, pak hodnota `\pdfimageresolution`.

Druhý řádek vloží na aktuální místo sazby horizontální box s naposledy načteným externím objektem. Pokud před `\pdfximage` neuvedeme `\immediate`, pak se objekt vloží do dokumentu až při referenci pomocí `\pdfrefximage`.

### Opakované použití obrázku

Někdy stejný obrázek zobrazujeme na více místech v jednom dokumentu. Typickým příkladem je logo v záhlaví každé stránky. Objektová struktura a systém referencí v PDF umožňuje odkázat na vložený objekt (např. jeho číslem). Protože PDF formát obsahuje tabulku s umístěním každého objektu v souboru, nemusíme se obávat časového zdržení při použití referovaného objektu způsobeného vyhledáváním. Uložíme-li si číslo objektu s vloženým obrázkem

```
\newtoks\regimgtoks
\immediate\pdfximage width.95\textwidth height5ex{drevo.jpg}
\edef\act{\regimgtoks={\the\pdflastximage}}\act
```

pak jej můžeme znovu použít třeba v patičce stránky

```
\def\img{\vbox to0pt{\hsize=.95\textwidth\vss
\hbox{\pdfrefximage\the\regimgtoks}}}
\makeatletter
\def\ps@plain{\def\@evenfoot{\thepage\hfil\img}%
\def\@oddfoot{\img\hfil\thepage}}
\makeatother
\pagestyle{plain}
```

Trochu složitější situace nastane, chceme-li obrázek opakovat v jiné velikosti, než jaká byla uvedena při vložení primitivem `\pdfximage`. Použijeme k tomu trik s dočasnou změnou transformační matice. Pro tento účel si na-





Obrázek 2: Simulace častého opakování obrázku. Referencemi na jednu vložené objekty lze několikanásobně zmenšit některé dokumenty. Například tento šestkrát (z 3 MB pouze 0,5 MB). *Obrázky v tomto článku pocházejí z publikace [4].*

definujeme dvě makra: první `\registimage<rule specification>{<file name>}` pro registraci obrázku (a jeho fyzické vložení do dokumentu) a druhé `\useimage{<file name>}{<desetinné číslo>}{<desetinné číslo>}` pro využití obrázku v konkrétním místě dokumentu. Ukažme si jejich použití na obrázku 2. Vznikl kódem:

```
\newdimen\tmpd \tmpd=4cm
\registimage height\tmpd {sum6th.jpg}
\def\defimg#1#2{\def#1{\useimage{sum6th.jpg}{#2}{#2}}}
\defimg\img{1} \defimg\imgg{.5} \defimg\imggg{.25}
\def\defquadro#1#2#3{\def#1{\vbox{\hsize=#3\tmpd
\halign{#####\cr#2&#2\cr#2&#2\cr}}}
{\tabskip=0pt\offinterlineskip \defquadro\qp\imgg{1}
\defquadro\qpp\imggg{.5} \defquadro\qppp\qpp{1}
\halign to\textwidth{\tabskip=0pt plus1fil#&##&\tabskip=0pt\cr
\img&\qp&\qpp\cr}}
```

A zde je definice obou maker:

```
\newbox\phantobox \setbox\phantobox=\hbox{
\def\registimage#1#2{\registimagee{#1}}
\def\registimagee#1#2{\immediate\pdfximage #1#2}%
\edef\act{\regimgtoks={\the\pdflastximage}}\act%
\expandafter\xdef\cname regimg:#2\endcsname{%
\pdfrefximage\the\regimgtoks}}
\def\useimage#1#2#3{f%
\setbox0=\hbox{\cname regimg:#1\endcsname}%
\wd\phantobox=#2\wd0
\ht\phantobox=#3\ht0 \dp\phantobox=#3\dp0
\wd0=0pt \ht0=0pt \dp0=0pt \leavevmode%
\pdfliteral{q #2 0 0 #3 0 0 cm}\box0\pdfliteral{Q}%
\copy\phantobox}}
```



Registrační makro `\registimage` ošetří nepovinný parametr typu  $\langle rule\ specification \rangle$  určující základní velikost obrázku. Trochu nezvyklá je druhá šachta v definici parametrů tohoto makra bez udání čísla parametru. Slouží jako separátor otevírací složené závorky [2, str. 39]. Výkonné makro `\registimagee` okamžitě vloží obrázek a jeho referenční číslo uloží do makra se jménem vytvořeným ze jména souboru.

Makro `\useimage` má tři parametry. Prvním je jméno souboru, který jsme registrovali makrem `\registimage`. Druhým a třetím jsou násobící koeficienty základní velikosti v horizontálním respektive vertikálním směru. Pracuje takto: nejprve do horizontálního boxu vloží odkaz na obrázek pomocí makra, definovaného při registraci pomocí jména souboru. Dále nastavíme velikost prázdného boxu `\phantombox` na velikost obrázku pronásobeného koeficientem zvětšení. Zvětšení/zmenšení provádíme transformační maticí PDF převádějící základní souřadnice do souřadnic stránky. Tuto transformaci popisuje maticová rovnice

$$\begin{bmatrix} x' & y' & 1 \end{bmatrix} = \begin{bmatrix} x & y & 1 \end{bmatrix} \begin{bmatrix} a & b & 0 \\ c & d & 0 \\ e & f & 1 \end{bmatrix}. \quad (1)$$

Neznámé hodnoty matice jsou v PDF zadané vektorem  $[a, b, c, d, e, f]$ . Hodnoty  $a$  a  $d$  uvádějí koeficienty zvětšení v horizontálním resp. vertikálním směru,  $e$  a  $f$  posun v těchto směrech a nenulové  $b$  a  $c$  způsobí sklonění a/nebo rotaci. Abychom zase nastavili původní transformaci, zapamatujeme si grafický stav pomocí operátoru  $q$  a vrátíme jej operátorem  $Q$ . Operátor transformační matice je  $cm$  a do PDF jej přidá primitivum `\pdfliteral`. Mezitím jsme vložili box s obrázkem, nulových rozměrů, aby se nám nerozjely referenční body PDF a  $\TeX$ u, který nemá ponětí o PDF transformaci. Proto na závěr podsuneme  $\TeX$ u prázdný box `\phantombox` o výsledných rozměrech obrázku.

## Závěr

Vkládat obrázky můžeme v pdf $\TeX$ u i pomocí makrobalíků, jejichž použití je jednodušší, než výše uvedené způsoby. Neumí však úplně vše, co je občas potřeba. Asi nejrozšířenější je  $\LaTeX$ ový `graphicx` Davida Carlisle a Sebastiana Rahtze, který podporuje všechny zmíněné formáty, rozlišuje PDF od EPS na základě typu výstupu, umí geometrické transformace – vše s uživatelsky přívětivým rozhraním klíč=hodnota. Makro pracuje i ve formátu plain, stačí postupně načíst soubory `miniltx.tex` a `graphicx.sty`.

Bohužel `graphicx` neumí vkládání stejného obrázku bez fyzického opakování. Pro příznivce Contextu dobrá zpráva: tuto vlastnost Hans Hagen do svého formátu vkomponoval. Uživatelé ostatních formátů musí, chtějí-li snížit velikost dokumentu, zatím použít způsob nastíněný v tomto článku.



## Reference

- [1] John D. Hobby. **A User's Manual for METAPOST**. Murray Hill, NJ 07974, 1995. Detailed.
- [2] Petr Olšák. **TeXbook naruby**. Konvoj Brno, 1. edition, 1997. Czech.
- [3] Hàn Thê Thành, Sebastian Rahtz, and Hans Hagen. **The pdfTeX user manual**, 2001. <http://www.tug.org/applications/pdftex/pdftex-s.pdf>.
- [4] Vít Zýka. **Živá a mrtvá Šumava aneb souznění slova a obrazu v čase**. Elektronická publikace na CD ROM ve formátu PDF. Dosud nevyšlo.

*Vít Zýka, zyka@cmp.felk.cvut.cz*

---

---

## Balíček fancyhdr.sty

DAVID MORAVEC

Článek popisuje, jak lze vytvářet uživatelskou podobu záhlaví a paty v dokumentech sázených systémem  $\text{\LaTeX} 2_{\epsilon}$ . Popisovaný balíček má mnoho možností použití. Ne všechny jsou zde však popsány. Pro podrobnější zkoumání doporučuji dokument [4].

### Předdefinované styly

Záhlaví a patu stránky můžeme v  $\text{\LaTeX}$ u měnit pomocí příkazu `\pagestyle`. Tento příkaz má jeden parametr, který obsahuje specifikaci stylu. Předdefinované styly jsou tyto:

**empty:** není vypisována žádná informace v záhlaví ani v patě.

**plain:** zobrazuje pouze číslo strany v patě. Záhlaví je prázdné. Toto je implicitní styl pro `article` a `report`.

**headings:** záhlaví obsahuje jméno kapitoly a číslo stránky. Pata je prázdná. Toto je implicitní styl pro `book`.

**myheadings:** pomocí příkazů `\markright` a `\markboth` lze zadat informace zobrazované v záhlaví.

Často se stane, že možnosti předcházejících příkazů uživateli nestačí. V tomto případě je vhodné použít balíček `fancyhdr`.