



OSSConf 2012: 91–96

NAUČÍM VÁS KRESLIŤ ALEBO PREDSTAVENIE BALÍČKA TikZ

ALEŠ KOZUBÍK (SK)

Abstrakt. Tento príspevok predstavuje balíček TikZ. Na obmedzenom priestore ilustruje elementárne nástroje tohto balíčka.

Kľúčové slová. L^AT_EX, TikZ, kresby.

I'LL TEACH YOU TO DRAW OR PRESENTATION OF THE TikZ PACKAGE

Abstract. This article brings the presentation of the TikZ package. On a limited space it introduces the very elementary tools of the package.

Keywords. L^AT_EX, TikZ, drawings.

Úvod

Ak sa už raz pustíme do sadzby dokumentov, skôr alebo neskôr nutne narazíme na potrebu vkladania grafických informácií do textu. Často sú to už hotové obrázky zhotovené pomocou špecializovaných grafických editorov alebo fotografie, ktoré jednoducho vkladáme na určené miesto v dokumente. No sú aj situácie, kedy žiadny zdroj potrebný grafický výstup neposkytuje a je teda na nás, aby sme si ho zhotovili takrečeno „svojpomocne“. V ďalšom texte sa predpokladá, že čitateľ má zvládnutú prácu so systémom T_EX, resp. L^AT_EX na úrovni príjemnej učebnice [4], podrobnej príručky [3] alebo aspoň zhutnených informácií z príspevkov [1] a [2].

Pri práci so systémom T_EX máme k dispozícii základnú sadu kresliacich nástrojov prostredia `picture`, ktorá umožňuje kreslenie jednoduchých aj zložitejších obrázkov, disponuje však pomerne obmedzenou sadou nástrojov a dosiahnutie požadovaného výsledku nemusí byť vždy jednoduché a užívateľsky pohodlné. Výhodou však je, že jeho výstupy dobre spolupracujú so všetkými drivermi. Ďalšou možnosťou je silný balík `pstricks`, je však špecializovaný na použitie postscriptovej grafiky a nie vždy spolupracuje s inými drivermi. Spomeňme aspoň vymenovaním niektoré ďalšie alternatívne balíčky ako sú `dratex`, `xypic`, `xfig`, alebo zo skupiny priateľov T_EXu program `metapost`.

My si v tomto príspevku v krátkosti predstavíme veľmi silný nástroj pre prácu s grafikou formátu `pgf` (Portable Graphics Format) v podobe balíčka TikZ (T ist kein Zeichenprogramm), ktorého autorom je Till Tantau. Manuál [6] tohto balíčka

je voľne dostupný napr. na adrese [8]. Nedajte sa odradiť jeho dĺžkou (takmer 730 strán) a majte vždy na pamäti, že manuál nie je váš nepriateľ.

Prv ako sa pustíme do experimentovania s balíčkom TikZ, treba si ešte povedať, že balíček dobre pracuje s výstupmi vo formátoch PostScript (dvips) a pdf (pdflatex, dvi_{pdf}mx), naopak dví prehliadače nemusia vždy výsledok zobrazovať korektne. Je preto potrebné pozeráť si vždy výstup ps alebo pdf.

1. Začínáme

Ak chceme pre svoje spracovanie obrázkov využívať nástroje z balíčka TikZ, musíme v prvom rade v preambule štandardným spôsobom jeho použitie deklarováť, teda vložiť riadky:

```
\usepackage{tikz}
\usetikzlibrary{arrows,shapes,trees,...}
```

Kým prvý riadok je nevyhnutný pre použitie balíčka TikZ, druhý riadok je nepovinný a slúži pre použitie niektorých špeciálnych rozširujúcich knižníc balíčka.

Pre samotné zaradenie obrázkov do textu máme dve možnosti. Prvou je príkaz `\tikz ... ;`, slúžiaci pre jednotlivé príkazy a zobrazenie výsledku na riadku. Tak napríklad príkazom `\tikz \draw[->] (0pt,0pt) -- (20pt,6pt);` vložíme do riadku šípku \longrightarrow . V prípade, že výsledok má byť vytvorený postupnosťou viacerých príkazov, uzavrieme ich do skupinových zátvoriek.

Pre vloženie obrázkov väčšieho rozsahu používame prostredie `tikzpicture`, teda:

```
\begin{tikzpicture}
...príkazy...
\end{tikzpicture}
```

2. Základné stavebné kamene






Činnosť makier z balíčka TikZ sa opiera o dva základné kamene. Sú to „cesty“ (path) a „uzly“ (node). Cestu môžeme charakterizovať ako postupnosť segmentov priamok a kriviek. Uzly potom môžeme označiť za prvky, ktoré sú pridávané ku cestám po ich vykreslení. Priblížme si teraz tieto základné dva prvky detailnejšie.

2.1. Paths čiže cesty

Cesta zadávaná príkazom `\path` nám slúži na vcelku ľubovoľné spájanie dvoch alebo viacerých bodov, pričom týmto spojniciam je možné pridávať mnohé vlastnosti. V zásade je možné jednotlivé cesty kresliť (`draw`), vyplňať vzniklé uzavreté tvary (`fill`) alebo vystrihovať pre ich následné použitie v ďalšom kreslení (`clip`). Vykreslenie čiary teda zadávame príkazom `\path[draw]`, čo je možné skrátiť na `\draw`, ak chceme vyplniť vzniklý útvar (zatiaľ nešpecifikovanou výplňou), použijeme `\path[fill]` alebo skrátene len `\fill` a konečne pre vystrihnutie určitej

časti `\path[clip]` alebo len `\clip`. Výsledky jednotlivých príkazov a ich kombinácií ilustruje tabuľka 1.

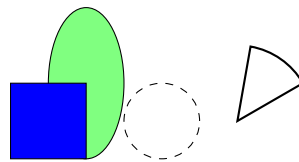
Tabuľka 1. Ukážky výsledkov základných príkazov pre kreslenie

	<code>\draw(1,1)--(1.5,1.5)--(2,1);</code>
	<code>\draw[line width=3pt](1,1)--(1.5,1.5)--(2,1)--cycle;</code>
	<code>\draw[fill=blue!30](1,1)--(1.5,1.5)--(2,1)--cycle;</code>
	<code>\fill[blue](1,1)--(1.5,1.5)--(2,1)--cycle;</code>
	<code>\draw[clip](1,1)--(1.5,1.5)--(2,1)--cycle;</code> <code>\fill[fill=red!50](1.5,1.6)circle(.5);</code>

Ako si možno všimnúť z tabuľky 1, jednotlivé príkazy majú rôzne možnosti voľby nepovinných argumentov. Ich význam v tu uvedených príkladoch je dostatočne intuitívny a kompletný zoznam možno nájsť v už spomenutom manuáli [6]. Všimnime si tiež, že každý príkaz je ukončený bodkočiarkou, čo je u príkazov balíčka TikZ povinné.

V balíčku TikZ sú preddefinované aj viaceré jednoduché geometrické tvary. Tak napríklad obdĺžnik vykreslíme príkazom `\draw(0,0)rectangle(2,1);`. Tu v prvej zátvorke uvádzame súradnice ľavého dolného rohu a v druhej súradnice pravého horného rohu. Vykreslenie kružnice so stredom v danom bode (a, b) a polomerom r docielime príkazom `\draw(a,b)circle(r);`. Konečne elipsu, ktorá je určená svojím stredom (s, t) a polosami (a, b) nakreslíme pomocou `\draw(s,t)ellipse(a and b);`. Použitie týchto príkazov a výsledok ilustruje nasledujúca ukážka.



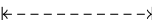



```
\begin{tikzpicture}
\draw[style=dashed](2,.5)circle(0.5);
\draw[fill=green!50](1,1)
ellipse(.5 and 1);
\draw[fill=blue](0,0)rectangle(1,1);
\draw[style=thick]
(3,.5)---+(30:1)arc(30:80:1)--cycle;
\end{tikzpicture}
```



V tejto ukážke si môžeme tiež všimnúť voliteľný parameter `style` príkazu `\draw`, ktorý upravuje vzhľad vykreslenej čiary na čiarkovanú, resp. tučnú. Posledný príkaz potom ilustruje možnosť použitia polárnych súradníc k vykresleniu časti oblúka `arc` kružnice, v tomto prípade v úseku od 30 po 80 stupňov.

Na záver tohto odstavca si ešte ukážeme niekoľko možností pre štýly kreslených čiar a možnosti vybavenia ich koncov šípkami rôznych tvarov. Pohľad do tabuľky 2 je určite dostatočne ilustratívny pre ich používanie.

Tabuľka 2. Ukážky rôznych štýlov kresliacich čiar

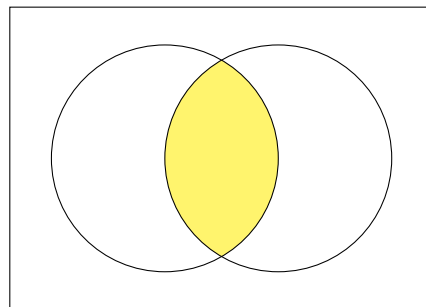
	<code>\draw[->] (0,0.5)--(2,0.5);</code>
	<code>\draw[thick,loosely dotted, >->] (0,0.5)--(2,0.5);</code>
	<code>\draw[thin,densely dashed, <->] (0,0.5)--(2,0.5);</code>
	<code>\draw[double,>-<] (0,0.5)--(2,0.5);</code>
	<code>\draw[very thick,latex-latex] (0,0.5)--(2,0.5);</code>
	<code>\draw[stealth-latex reversed,dashed] (0,0.5)--(2,0.5);</code>

2.2. Vystrihovanie čiže clip

Voľný preklad anglického *clipping* do slovenčiny ako vystrihovanie nezodpovedá celkom presne tomu, čo sa pri použití príkazu `\clip` deje. Po použití príkazu `\clip` sú zo všetkých kresieb nasledujúcich po ňom „vystrihnuté“ len tie časti, ktoré ležia vo vnútri útvaru vykresleného pri použití `\clip`. Vysvetlíme si to na príklade v poslednom riadku tabuľky 1. Najskôr sme príkazom `\draw[clip]` „vystrihli“ trojuholník so zadanými vrcholmi. V ďalšom kroku sa už preto zobrazila len tá časť vyplneného kruhu, získaného pomocou `\fill`, ktorá leží vo vnútri tohto trojuholníka.

Trvalé vystrihnutie určitej oblasti z obrázku môže byť veľmi nepraktické. Preto máme k dispozícii prostredie `scope`, ktorým je možné obmedziť platnosť príkazu `\clip` iba v rámci tohto prostredia. Ukážeme si to na príklade zobrazenia prieniku dvoch množín. Tu najskôr do prostredia `scope` uzavrieme oblasť ohraničenú oblúkmi dvoch kružníc, ktorú následne vyfarbíme. Po ukončení prostredia `scope` pokračujeme vo vykreslení zvyšku oboch kružníc.

```
\begin{tikzpicture}
\draw (-2.8, 2) rectangle (2.8, -2);
\begin{scope}
\clip (-0.75, 0) circle (1.5);
\clip ( 0.75, 0) circle (1.5);
\fill[color=yellow!70] (-2,1.5)
rectangle (2,-1.5);
\end{scope}
\draw (-0.75, 0) circle (1.5);
\draw ( 0.75, 0) circle (1.5);
\end{tikzpicture}
```



2.3. Uzly čiže nodes

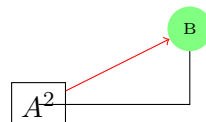
Uzly sú pridávané ku cestám po ich vykreslení. Pre ďalšie odkazovanie na tieto uzly pri tvorbe obrázka im môžeme priradiť aj mená. Jedným z praktických využití uzlov je pridávanie popiskov do obrázka. Tak napríklad lomenú čiaru, ktorej konce označíme písmenami môžeme vykresliť takto:

```
\begin{tikzpicture}
\draw (0,0) node {A}--(2,0)--(2,0.5) node {B};
\end{tikzpicture}
```



Ak by sme ešte chceli následne spojiť body A a B šípkou, využijeme možnosť uzly pomenovať. To prezentuje nasledujúca ukážka.

```
\begin{tikzpicture}
\draw (0,0) node[draw] (nodeA){ $A^2$ }--(2,0)--(2,1)
node[circle,fill=green!50] (nodeB) {\tiny B};
\draw[red,->] (nodeA)--(nodeB);
\end{tikzpicture}
```



Výhodnosť použitia pomenovaných uzlov môžeme pozorovať na červenej šípke, ktorá je viditeľná len po hranice objektov vykreslených v uzloch. Ak by sme sa spoliehali len na súradnice, prekryla by aj tieto objekty, čo by bolo v tomto prípade nežiadúce.

Každý uzol má okrem pomenovania a zadanej pozície dva parametre. Jedným je tvar a druhým textová značka. Tak sme v predchádzajúcej ukážke prideliť uzlu B tvar kruhu so zelenou výplňou a textovú značku B vysádzanú drobným písmom. Zároveň na obrázku pozorujeme, že umiestnenie písmena A je nevhodné a koliduje s vodorovnou čiarou, vychádzajúcou z tohto bodu. Tomu je možné zabrániť vhodným umiestnením textovej skupiny na vhodnú pozíciu relatívne ku polohe uzla. Nasledujúca ukážka ilustruje rozmiestnenie štyroch písmen na štyri základné pozície vzhľadom na polohu uzla vyznačeného bodkou.

```
\fill (0,2) circle (1pt) node[below] {$A$}
node[right] {$B$} node[above] {$C$}
node[left] {$D$};
```



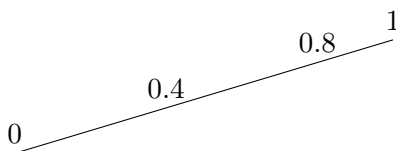
Polohu popisiek umiestnených do daného uzla je možné určiť aj kombinovaním štyroch základných polôh. Ilustruje to nasledujúca ukážka:

```
\fill (0,2) circle (1pt) node[below left] {$A$}
node[below right] {$B$} node[above right] {$C$}
node[above left] {$D$};
```



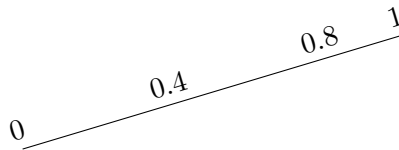
Na záver si ešte ukážeme, ako je možné umiestniť text do uzlov rozmiestnených pozdĺž vykreslenej cesty. K tomu slúži voliteľný parameter `pos` príkazu `\node`. Tento parameter nadobúda hodnoty medzi 0 a 1 a určuje tak relatívnu pozíciu uzlu medzi začiatkom a koncom vykreslenej cesty. Ukážka ilustruje rozmiestnenie štyroch uzlov pozdĺž cesty.

```
\draw (0,0)--(5,1.5)
node[pos=0,above]{0}
node[pos=.4,above]{0.4}
node[pos=0.8,above]{0.8}
node[pos=1,above]{1};
```



Ako vidíme z obrázka, poloha niektorých popisov nie je príliš estetická. To sa prejavuje u hodnôt 0.4 a 0.8, ktoré ako by smerom dopredu odstávali od vykreslenej priamky. Tento problém je potom možné odstrániť pridaním voliteľného parametra `sloped`, ktorý má za následok prispôsobenie sklonu písma ku sklonu vykreslenej cesty. Výsledok vidíme tu:

```
\draw (0,0)--(5,1.5)
  node[pos=0,above,sloped]{0}
  node[pos=.4,above,sloped]{0.4}
  node[pos=0.8,above,sloped]{0.8}
  node[pos=1,above,sloped]{1};
```



3. Záver

V príspevku sme si na obmedzenom priestore predstavili elementárne nástroje, ktoré môžu slúžiť pre potreby grafickej prezentácie obsahu. Akým silným nástrojom je balíček `TikZ` sa záujemcovia môžu presvedčiť napr. na stránke [7], ktorá obsahuje bohatú škálu príkladov, či už v podobe výsledných dokumentov formátu `pdf` alebo aj v podobe zdrojového kódu pre `LATEX`.

Literatúra

- [1] BLAŠKO, R.: *L^AT_EX nie je farba na maľovanie*, Zborník príspevkov medzinárodnej konferencie OSSConf 2010, 1.–4. júla 2010, Žilina, str. 43–52. ISBN 978-80-970457-0-8.
- [2] BLAŠKO, R.: *L^AT_EX nie je farba na maľovanie, ale na písanie*, Zborník príspevkov medzinárodnej konferencie OSSConf 2011, 6.–9. júla 2011, Žilina, str. 223–236. ISBN 978-80-970457-1-5.
- [3] KOPKA, H. – DALY, P.W.: *L^AT_EX – podrobný príručce*, Computer Press, Brno, 2004. ISBN 80-722-6973-9.
- [4] RYBIČKA, J.: *L^AT_EX pro začátečníky*, KONVOJ, Brno, 2003. ISBN 80-7302-049-1.
- [5] RYBIČKA, J. – ČAČKOVÁ, P. – PŘICHYSTAL, J.: *Průvodce tvorbou dokumentů*, NAKLADATELSTVÍ MARTIN STRÍŽ, Bučovice, 2011. ISBN 80-7302-049-1.
- [6] TANTAU, T.: *The TikZ and PGF Packages Manual for version 2.10*, <http://sourceforge.net/projects/pgf>.
- [7] <http://www.texample.net>, sample resources for T_EX users.
- [8] <http://www.ctan.org/tex-archive/graphics/pgf/base/doc/generic/pgf/pgfmanual.pdf>.

Kontaktná adresa

RNDr. Aleš Kozubík, PhD., Katedra matematických metód, Fakulta riadenia a informatiky, Žilinská univerzita, Univerzitná 8215/1, 010 26 Žilina, Slovenská Republika,
E-mailová adresa: alesko@frcatel.fri.uniza.sk