

Zpravodaj Československého sdružení uživatelů TeXu

Miroslav Dont

AmSLaTeX verze 1.0 a 1.1

Zpravodaj Československého sdružení uživatelů TeXu, Vol. 1 (1991), No. 4, 13–19

Persistent URL: <http://dml.cz/dmlcz/148816>

Terms of use:

© Československé sdružení uživatelů TeXu, 1991

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>

$e_1 a_2, u_1, 2, \dots$

a výnimiek do `\hyphenation{médium, ...}`. Tejto časti však nebola zatiaľ venovaná veľká pozornosť.

Čo dodať na záver: Snáď zodpovedá aktuálnu otázku „Ako bude rozdelené (slovo) Československo?“ Vytvoreným slovenským slovníkom: Čes-ko-slo-ven-sko ...

Literatura

- [1] Knuth D. E., The \TeX book, Addison-Wesley Pub. Comp. A AMS, 1986.
- [2] Oravec J. a Laca V., Príručka slovenského pravopisu pre školy, SPN, Bratislava, 1976.
- [3] Mistrík J., Frekvencia tvarov a konštrukcií v slovenčine, VEDA, Bratislava, 1985.

(Jana Chlebíková)

$\mathcal{A}\mathcal{M}\mathcal{S}$ - \LaTeX verze 1.0 a 1.1

V červenci roku 1990 nabídla American Mathematical Society veřejnosti první verzi $\mathcal{A}\mathcal{M}\mathcal{S}$ - \LaTeX u. Tato verze byla označena jako 1.0. V současné době je k dispozici verze 1.1, která se však od verze 1.0 liší zcela nepodstatně — ve verzi 1.1 byly pouze opraveny některé známé chyby. Dále budu proto jednoduše mluvit o $\mathcal{A}\mathcal{M}\mathcal{S}$ - \LaTeX u.

Pokud nepočítám příznivce PLAINu a odpůrce velkých balíků maker vůbec, stojí každý budoucí uživatel \TeX u před volbou, zda používat $\mathcal{A}\mathcal{M}\mathcal{S}$ - \TeX nebo \LaTeX (volba \LaTeX u je samozřejmě ta nejlepší!). Pravděpodobně ve snaze tuto volbu ulehčit, byly činěny různé pokusy, jak $\mathcal{A}\mathcal{M}\mathcal{S}$ - \TeX přiblížit \LaTeX u nebo naopak. Výsledkem nejznámějších dvou z těchto pokusů jsou balíky \LaTeX - \TeX a $\mathcal{A}\mathcal{M}\mathcal{S}$ - \LaTeX . Můj soukromý dojem však je, že uživatelé \LaTeX u nebo $\mathcal{A}\mathcal{M}\mathcal{S}$ - \TeX u se pouze upevní ve víře, že jejich volba byla ta správná. Je třeba zdůraznit, že \LaTeX - \TeX a $\mathcal{A}\mathcal{M}\mathcal{S}$ - \LaTeX není jedno a totéž a tyto systémy jsou dokonce nekompatibilní. Systém \LaTeX - \TeX je rozšíření $\mathcal{A}\mathcal{M}\mathcal{S}$ - \TeX u, což mimo jiné vychází ze skutečnosti, že autorem \LaTeX - \TeX u je M. D. Spivak (tj. autor $\mathcal{A}\mathcal{M}\mathcal{S}$ - \TeX u). \LaTeX - \TeX přijímá mnohé rysy \LaTeX u jako např. automatické číslování formulí, kapitol, sekcí atd., možnost křížových referencí, okolí pro snadnou tvorbu tabulek atd. Věc, kterou \LaTeX - \TeX od \LaTeX u nepřebírá, je okolí `picture` (okolí pro tvorbu jednoduchých obrázků přímo ve zdrojovém textu).

Na druhé straně dává \LaTeX - \TeX velké možnosti při vytváření komutativních diagramů; kdo v textu potřebuje mnoho komutativních diagramů, tomu bych doporučil \LaTeX - \TeX . Další výhodou \LaTeX - \TeX u je poměrná uzavřenost systému a podrobná dokumentace (M. D. Spivak, který již dříve napsal *The Joy of \TeX* — manuál $\mathcal{A}\mathcal{M}\mathcal{S}$ - \TeX u — napsal podrobný manuál pro \LaTeX - \TeX).

Na rozdíl od \LaTeX - \TeX u je $\mathcal{A}\mathcal{M}\mathcal{S}$ - \LaTeX rozšířením \LaTeX u. Přitom rozšířením v tom smyslu, že základem je dokonce přímo soubor `latex.tex` bez nějakých úprav (předpokládá se, že bude použit soubor `latex.tex` verze 2.09 a to ne starší než z května 1986). Tato skutečnost mimo jiné znamená, že v $\mathcal{A}\mathcal{M}\mathcal{S}$ - \LaTeX u budou k dispozici všechny příkazy \LaTeX u tak, jak je uživatel zná z L^amportova manuálu (pouze příkazy pro volbu fontů budou mít poněkud jiný význam jak se zmíním dále). Kdo

má zkušenosti s L^AT_EXem, může použít $\mathcal{A}\mathcal{M}\mathcal{S}$ -L^AT_EX téměř bez jakéhokoli dalšího učení — pouze stačí prolistovat stručnou uživatelskou příručku (která je dodávána spolu s ostatními soubory jako L^AT_EXový soubor a je možné si ji vytisknout ještě před instalací $\mathcal{A}\mathcal{M}\mathcal{S}$ -L^AT_EXu).

Balík obsahující $\mathcal{A}\mathcal{M}\mathcal{S}$ -L^AT_EX se skládá v podstatě ze dvou částí. Jedna část se skládá z velkého množství .STY-souborů a je vlastní nadstavbou L^AT_EXu — o těchto souborech se zmíním dále. Druhá část se týká především způsobu výběru fontů a tvoří pevnou část $\mathcal{A}\mathcal{M}\mathcal{S}$ -L^AT_EXu (zatímco nové stylové soubory uživatel může ale nemusí v $\mathcal{A}\mathcal{M}\mathcal{S}$ -L^AT_EXu používat, změna schématu volání fontů bude zabudována přímo v novém .fmt souboru). Podívejme se nyní stručně na tuto pevnou část $\mathcal{A}\mathcal{M}\mathcal{S}$ -L^AT_EXu. Nejedná se o nic jiného než o NFSS (new font selection scheme) autorů F. Mittelbacha a R. Schöpfy, které je též k dispozici k samostatnému použití se standardní verzí L^AT_EXu a které by mělo být součástí budoucí nové verze L^AT_EXu (z tohoto důvodu se mi zdá být užitečné seznámit se s NFSS co nejdříve). Základní myšlenka NFSS spočívá v tom, že příkazy pro přepínání fontů v NFSS mění pouze jeden atribut písma, což umožní použít většího množství fontů při stejném počtu příkazů. Teoreticky může samozřejmě uživatel používat všechny fonty, které má k dispozici, pokud si ovšem nadefinuje přístup k těmto fontům - zvláště pro začátečníka to může být nepříjemná záležitost. Ve standardním L^AT_EXu příkazy pro změnu fontů jako třeba `\bf`, `\it`, `\sc`, atd. způsobí vždy kompletní změnu fontu. Pokud jsem např. použil přepínače `\sl`, následující text bude tištěn „slanted“ (tj. nakloněným) písmem. Pokud dále použiji přepínač `\bf`, bude následující text tištěn písmem tučným, ale nikoli již nakloněným — přepínač `\bf` přepne tedy nejen na písmo tučné ale zároveň z nakloněného na „normální stojaté“. V NFSS jsou rozlišovány tři atributy písma, které se přepínají každé zvlášť. Jsou to atributy *shape*, *series* a *size*. Příkaz `\size` mění velikost písma (a zároveň vzdálenost řádků). Příkaz `\shape` mění tvar písma (jako např. *italic*, *small caps*, *slanted*). Příkaz `\series` mění „tučnost“ písma, ale nemění velikost ani tvar. Přitom „tučnost“ je možné nastavovat v poměrně široké škále a přitom je možné nastavovat zvlášť „tučnost ve vodorovném směru“ a „tučnost ve svislém směru“ (pokud jsou k dispozici příslušné fonty). Dále je tu ještě příkaz `\family`, který volí základní fontový rámec (základní skupinu, kde se fonty liší pouze třemi výše uvedenými atributy; nebudu zde tyto pojmy rozebírat podrobně — odkazuji na článek F. Mittelbacha a R. Schöpfy o NFSS z TUGboatu č. 2 z r. 1990). Uvedené příkazy většinou uživatel nepoužívá přímo, ale jsou použity při tvorbě příkazů `\rm`, `\bf`, `\it`, `\sl`, `\sc`, které mají tedy jiný význam než ve standardním L^AT_EXu. Tak např. použijeme-li nyní po sobě příkazy `\sl\bf` (nebo také v obráceném pořadí), bude dále tištěno tučné nakloněné písmo (opět pokud bude k dispozici příslušný font).

Jak jsem již poznamenal, při instalaci $\mathcal{A}\mathcal{M}\mathcal{S}$ -L^AT_EXu je třeba vyrobit nový .fmt soubor, avšak použije se přitom starého souboru `latex.tex` (a starého souboru `lplain.tex`). Místo souboru `lfonts.tex` se však použije soubor `lfonts.new`, který je dodáván v rámci $\mathcal{A}\mathcal{M}\mathcal{S}$ -L^AT_EXu. V návodu se doporučuje následující postup. Starý soubor `lplain.fmt` přejmenujeme (pro případné další použití standardní verze L^AT_EXu). Dále odstraníme nebo přejmenujeme soubor `lfonts.tex` a nahradíme souborem `lfonts.new` (zároveň umístíme další dodávané soubory do příslušných adresářů). Nyní vyrobíme nový soubor `lplain.fmt` — pomocí původního souboru `lplain.tex`. V určitém okamžiku se inicializace zastaví a objeví se hlášení, že INITEX nemůže nalézt soubor `lfonts.tex` a dotaz na nový soubor — napíšeme `lfonts.new`.

Během dalšího průběhu se objeví dotazy na některé další soubory a odpovědi budou záviset na tom, jaké fonty budeme chtít používat a jak mnoho jich chceme mít v paměti během práce \TeX u. Např. po dotazu na soubor `fontdef.tex` máme možnost tří odpovědí, podle toho, zda chceme pracovat pouze s CM fonty a \LaTeX ovými fonty nebo zda chceme pracovat se základními matematickými fonty z kolekce AMS-fontů nebo zda chceme používat všechny AMS-fonty. Jak vidíme, $\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\text{\LaTeX}$ umožňuje mimo jiné pracovat s AMS-fonty — možnost práce s AMS-fonty však není důvod proč používat $\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\text{\LaTeX}$.

Jak jsem již poznamenal, další část $\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\text{\LaTeX}$ u je velký počet `.STY` souborů (přitom lze používat všechny `.STY` soubory, které jsou k dispozici pro původní verzi \LaTeX u). Základní stylové soubory jsou `amstex.sty`, `amsart.sty` a `amsbook.sty`. Soubory `amsart.sty` a `amsbook.sty` jsou samostatné stylové soubory, které jsou analogií standardních \LaTeX ových souborů `article.sty` a `book.sty` (podobně jako k těmto souborům, existují i k souborům `amsart.sty` a `amsbook.sty` pomocné soubory umožňující snadný tisk v 10-ti, 11-ti a 12-ti bodovém písmu. Podstatný je však soubor `amstex.sty`, který nepoužíváme jako samostatný stylový soubor, ale jako volbu, chceme-li pracovat s příkazy (především příkazy v `math`-modu), které znají uživatelé $\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\text{\TeX}$ u, ale které nejsou definovány ve standardním \LaTeX u. Chce-li uživatel pracovat s těmito příkazy a přitom psát soubor např. ve standardním \LaTeX ovém stylu `article`, použije v záhlaví příkaz

```
\documentstyle[amstex]{article}
```

Soubor `amstex.sty` je přitom modifikovaná verze souboru `amstex.tex` z $\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\text{\TeX}$ u verze 2.0. Je třeba říci, že příkazy zde definované nejsou zcela identické s příkazy v $\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\text{\TeX}$ u, ale jsou svou syntaxí přizpůsobeny syntaxi užívané v \LaTeX u. Mimo jiné to znamená, že se zde používá `\begin{\dots}` a `\end{\dots}` stejně jako v \LaTeX u, tj. např. místo

```
\matrix
...
\endmatrix
```

(což bychom použili v $\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\text{\TeX}$ u) zde budeme psát

```
\begin{matrix}
...
\end{matrix}
```

stejně jako u jiného okolí v \LaTeX u. Zmíním se zde pouze velice stručně o některých příkazech definovaných v souboru `amstex.sty`.

Mezerování v `math`-modu je stejné jako v \LaTeX u, tj. pomocí příkazů `\`, `\:` `\;` a `\!`; jsou zde ale navíc příkazy `\@`, a `\@!`, které vyrobí kladné nebo záporné mezery, jež jsou desetinou mezer generovaných příkazy `\`, nebo `\!` (nevím dost dobře situaci, kdy by bylo skutečně potřeba takto jemné mezerování — ostatně v takovémto případě by bylo možné příslušnou mezeru vyrobit různými jinými způsoby). Stejně jako v $\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\text{\TeX}$ u jsou definovány příkazy pro dvojný, trojný a čtyřnásobný integrál a také pro znaky integrálu se třemi tečkami mezi nimi (tyto příkazy uživatel přivítá

vzhledem k tomu, že mu odpadne starost o správné mezerování mezi integrály). Pokud se týče tří teček, uživatel \LaTeX u je zvyklý na příkazy `\ldots` a `\cdots` (a dále `\vdots` a `\ddots` pro použití v maticích). V souhlase s $\mathcal{M}\mathcal{S}\text{-}\text{\TeX}$ em bude nyní příkaz ... umisťovat tři tečky buď na řádku nebo na střed řádky podle toho, co je za tímto příkazem. Dále stejně jako v $\mathcal{M}\mathcal{S}\text{-}\text{\TeX}$ u fungují příkazy jako `\dotsc`, `\dotscsb` atd.; \LaTeX ové příkazy `\ldots`, `\cdots` jsou zachovány.

Oproti \LaTeX u je podstatně vylepšeno umisťování dvojitých akcentů v mathmodu a funguje stejně jako v $\mathcal{M}\mathcal{S}\text{-}\text{\TeX}$ u. \LaTeX ový příkaz `\stackrel` je zde vylepšen příkazy `\overset` a `\underset`, které umožňují umisťovat symboly nad nebo pod binární relace a dále příkazem `\sideset`, který umožní umisťovat symboly do rohů velkých operátorů. Vedle \LaTeX ového příkazu `\frac` jsou zde příkazy `\dfrac`, `\tfrac`, což jsou zkratky pro

```
{\displaystyle\frac ... } a {\textstyle\frac ... };
```

dále může být explicitně nastavena tloušťka lomítkové čáry. Stejně jednoduše jako v $\mathcal{M}\mathcal{S}\text{-}\text{\TeX}$ u lze generovat řetězové zlomky.

V \LaTeX u se k psaní matic většinou používá okolí `array`. Toto okolí je v $\mathcal{M}\mathcal{S}\text{-}\text{\LaTeX}$ u zachováno, ale z $\mathcal{M}\mathcal{S}\text{-}\text{\TeX}$ u jsou převzaty příkazy `\matrix`, `\pmatrix`, `\bmatrix`, `\vmatrix`, `\Vmatrix` (které jsou zde definovány jako okolí) a dále `\smallmatrix`.

Podstatně je oproti \LaTeX u rozšířena možnost psaní víceřádkových formulí. Zatímco v \LaTeX u se k tomuto účelu používalo okolí `eqnarray` (`eqnarray*`) nebo `array` (kde již bylo dost nepřijemné, že často bylo potřeba v každé kolonce psát příkaz `\diplaystyle` a bylo potřeba doladit mezerování mezi řádky), jsou v $\mathcal{M}\mathcal{S}\text{-}\text{\LaTeX}$ u k dispozici např. okolí

<code>align</code>	<code>gather</code>	<code>alignat</code>	<code>multline</code>
<code>split</code>	<code>xalignat</code>	<code>xxalignat</code>	

Nebudu zde tato okolí popisovat podrobně. Chceme-li vsunout řádek nebo několik řádků textu mezi řádky takto zarovnané formule, můžeme jednoduše použít příkaz `\intertext` (v \LaTeX u bylo něco takového pro uživatele podstatně komplikovanější). Číslování formulí je možné snadno nadefinovat v různém stylu (např. průběžné číslování, číslování v rámci kapitoly nebo sekce atd.).

Zmínil jsem se pouze o některých příkazech, které jsou definovány v souboru `amstex.sty`. Podrobněji jsou příkazy popsány v uživatelské příručce, která je spolu s $\mathcal{M}\mathcal{S}\text{-}\text{\LaTeX}$ em dodávána v elektronické podobě. Jelikož se jedná o soubor psaný v \LaTeX u (aby bylo možné příručku vytisknout ještě před instalací $\mathcal{M}\mathcal{S}\text{-}\text{\LaTeX}$ u), nejsou zde ilustrovány některé příkazy na příkladech. Takové příklady je možné nalézt v souborech `testart.tex` a `testbook.tex` (k těmto souborům existují některé další pomocné, které jsou potřeba během zpracování), které jsou již psány v $\mathcal{M}\mathcal{S}\text{-}\text{\LaTeX}$ u.

Mimo stylový soubor `amstex.sty` zde jsou některé další, které lze také přidávat jako volitelné stylové soubory. Je to např. soubor `amscd.sty`, který umožňuje snadné psaní komutativních diagramů; makra pro tento účel jsou umístěna do samostatného stylového souboru proto, aby se ušetřilo alespoň trochu paměti (o tom, jak je to s pamětí při práci s $\mathcal{M}\mathcal{S}\text{-}\text{\LaTeX}$ em se ještě zmíním dále). Psaní komutativních diagramů je zde stejně jako v $\mathcal{M}\mathcal{S}\text{-}\text{\TeX}$ u (až na to, že místo

```
\CD ... \endCD
```

píšeme

```
\begin{CD} ... \end{CD}).
```

V *podstatě* stejné diagramy lze psát bez těchto maker ve standardním L^AT_EXu např. s použitím okolí `array`. Následující příklad je převzat z uživatelské příručky $\mathcal{A}\mathcal{M}\mathcal{S}$ -L^AT_EXu (soubor `amslatex.tex`). Diagram tvaru

$$\begin{array}{ccc} S^{W_\Lambda} \otimes T & \xrightarrow{j} & T \\ \downarrow & & \downarrow \text{End } P \\ (S \otimes T)/I & \xlongequal{\quad} & (Z \otimes T)/J \end{array}$$

lze ve standardním L^AT_EXu vyrobit tímto způsobem:

```
\begin{array}{ccc}
\begin{array}{c} S^{\{\cal W\}_\Lambda} \otimes T \\ \stackrel{j}{\longrightarrow} \\ \Big\downarrow \end{array} & & \begin{array}{c} T \\ \downarrow \text{End } P \end{array} \\
(S \otimes T)/I & \xlongequal{\quad} & (Z \otimes T)/J
\end{array}
```

Přitom v $\mathcal{A}\mathcal{M}\mathcal{S}$ -L^AT_EXu při použití maker ze souboru `amscd.sty` napíšeme analogický diagram příkazy

```
\begin{CD}
S^{\{\cal W\}_\Lambda} \otimes T @>j>> T \\
@VVV @VV\text{End } P\downarrow V \\
(S \otimes T)/I @= (Z \otimes T)/J
\end{CD}
```

(zde se předpokládá, že příkaz `\End` byl definován pomocí $\mathcal{A}\mathcal{M}\mathcal{S}$ -L^AT_EXového příkazu `\operatorname{End}`) — v L^AT_EXu bychom mohli použít příkaz `\mathop`. Takto generovaný diagram nebude totožný se zde uvedeným; vodorovná šipka bude o něco delší a mezerování kolem „rohů“ diagramu bude lepší. Daná makra umožňují tvorbu pouze „obdélníkových“ diagramů neumožní tedy např. napsat diagram, který by obsahoval šikmé šipky. Složitější diagramy (např. tedy se šikmými šípkami) by bylo možné vytvořit s použitím L^AT_EXového okolí `picture`. Tento způsob je však poměrně pracný a výsledek nemusí být vždy uspokojivý. Jak jsem již poznamenal, dobrá makra pro psaní nejrozumnějších typů diagramů lze nalézt v $\mathcal{L}\mathcal{A}\mathcal{M}\mathcal{S}$ -T_EXu.

Soubor `amscd.sty` lze používat buď samostatně nebo spolu se souborem `amstex.sty`. Pokud jej používáme spolu se souborem `amstex.sty`, je potřeba jej samostatně volat, tj. použít např. příkaz `\documentstyle[amstex,amscd]{article}`

apod. Soubor `amstex.sty` přitom automaticky volá některé jiné soubory — jsou to soubory

`amsfonts.sty`, `amsbsy.sty` a `amstext.sty`

(poznamenejme, že pokud použijeme `amsart.sty` nebo `amsbook.sty`, pak je automaticky volán `amstex.sty`). Tyto soubory mohou být ovšem použity také samostatně. Soubor `amsfonts.sty` a dále také soubor `amssymb.sty` je určen pro práci s AMS-fonty (verze 2.0). Soubor `amsfonts.sty` obsahuje mimo jiné definici příkazu `\newsymbol` který umožňuje pracovat se speciálními symboly AMS-fontů a který je popsán v uživatelské příručce AMS-fontů. Pokud používáme tyto symboly často, není třeba pokaždé definovat příslušné příkazy pomocí `\newsymbol`, ale je možné použít soubor `amssymb.sty`, který obsahuje definice jmen všech těchto asi 200 symbolů. Ostatně při instalaci $\mathcal{A}\mathcal{M}\mathcal{S}$ - $\mathcal{L}\mathcal{A}\mathcal{T}\mathcal{E}\mathcal{X}$ u je možné tyto definice zahrnout přímo do `.fmt` souboru. Soubor `amsbsy.sty` obsahuje definice příkazů `\boldsymbol` a `\pmb`. Příkazem `\boldsymbol` lze tisknout tučné matematické symboly (pokud existují příslušné fonty). Např. pří-

kazy `\boldsymbol{\infty}`, `\boldsymbol{\pi}` nebo `\boldsymbol{+}`

vytisknou tučné znaky ∞ , π a $+$. Příkaz `\pmb` („poor man’s bold“) použijeme k tištění tučných znaků, pro které není možné použít příkaz `\boldsymbol`. Tento příkaz pracuje tak, že tiskne třikrát trochu posunutý tentýž znak a vytvoří tak zdání tučného symbolu. Nedoporučuje se ale používat spolu s velkými operátory jako třeba suma a pod. (vzhledem ke špatnému umístování symbolů pod nebo nad operátorem). Soubor `amstext.sty` obsahuje definici jediného příkazu a to `\text` převzatého z $\mathcal{A}\mathcal{M}\mathcal{S}$ - $\mathcal{T}\mathcal{E}\mathcal{X}$ u. Funguje podobně jako `\mbox` v $\mathcal{L}\mathcal{A}\mathcal{T}\mathcal{E}\mathcal{X}$ u, až na to, že při použití `\mbox`u v indexu se nemění velikost písma uvnitř `\mbox` u, zatímco při použití příkazu `\text` — ano. Zatímco výše uvedené stylové soubory lze používat i samostatně bez `amstex.sty`, některé další lze používat pouze spolu se souborem `amstex.sty` (přitom `amstex.sty` je nevolá automaticky). Jsou to soubory

`intlim.sty` `nonamelm.sty` `nosumlim.sty` `righttag.sty` `ctagsplt.sty`

Použití těchto souborů má následující význam (názvy jsou ostatně samovyšvětlující). Při použití souboru `intlim.sty` budou meze u integrálů tištěny pod a nad integrál, nikoli vedle integrálu. Při použití `nosumlim.sty` budou naopak meze u znamení sumace tištěny vedle \sum . Podobně při použití souboru `nonamelm.sty` budou meze operátorů definovaných pomocí příkazu `\operatorname` tištěny vedle operátorů a ne pod nebo nad. Soubor `righttag.sty` způsobí, že číslování rovnic bude umístováno vpravo místo vlevo. Nakonec při použití souboru `ctagsplt.sty` bude rovnice generovaná pomocí okolí `split` očíslována a číslování bude vertikálně centrováno vzhledem k této rovnici (na rozdíl od ostatních okolí používaných pro zarovnávání rovnic, okolí `split` normálně rovnici nečísluje — pro ostatní okolí existuje ale vždy *-verze, která také nečísluje).

U většiny souborů, které distribuuje $\mathcal{A}\mathcal{M}\mathcal{S}$ v rámci $\mathcal{A}\mathcal{M}\mathcal{S}$ - $\mathcal{L}\mathcal{A}\mathcal{T}\mathcal{E}\mathcal{X}$ u, není uváděn autor souboru. Výjimku tvoří soubory související s NFSS, jejichž autoři jsou F. Mittelbach a R. Schöpf. Od stejných autorů jsou soubory `theorem.sty` a `verbatim.sty`, které jsou rovněž dodávány spolu s $\mathcal{A}\mathcal{M}\mathcal{S}$ - $\mathcal{T}\mathcal{E}\mathcal{X}$ em (a několik dalších souborů, které souvisí s těmito soubory). Soubor `verbatim.sty` obsahuje definici okolí `verbatim`, což je vylepšená verze okolí stejného jména z $\mathcal{L}\mathcal{A}\mathcal{T}\mathcal{E}\mathcal{X}$ u, které slouží k doslovnému tisku části textu (např. výše uvedené ukázky příkazů, kterými byl tištěn příklad komutativního diagramu, byly generovány pomocí $\mathcal{L}\mathcal{A}\mathcal{T}\mathcal{E}\mathcal{X}$ ového okolí `verbatim` — stačilo pouze

vsunout příslušný blok mezi `\begin{verbatim}` a `\end{verbatim}`). Dále je v souboru `verbatim.sty` definováno okolí `comment`, které má ten efekt, že všechn text mezi `\begin{comment}` a `\end{comment}` je při zpracování \TeX em ignorován. Stylový soubor `theorem.sty` poskytuje mnohem více možností pro definování okolí typu **Věta**, **Tvrzení**, **Definice** atd. Tyto soubory (`theorem.sty` a `verbatim.sty`) je možné používat samostatně a to dokonce v rámci standardního \LaTeX u. Původně tyto soubory byly skutečně napsány jako stylové soubory pro standardní \LaTeX (ze stylových souborů autorů Mittelbacha a Schöpf je u nás asi nejznámější styl `multicolumn`, který umožňuje paní víceloupcového textu). Soubor `theorem.sty` bude automaticky volán při použití `amsart.sty` nebo `amsbook.sty`.

Už při používání standardního \LaTeX u se můžeme setkat s potížemi s pamětí např. při vícenásobném použití příkazu `\bezier` na jedné stránce nebo při mnohonásobném použití příkazu `\label` (návěští pro křížové reference) nebo u delších textů dokonce jenom proto, že je příliš mnoho příkazů `\section` apod. V $\mathcal{AMS}\text{-}\text{\LaTeX}$ u, který kombinuje \LaTeX s mnoha příkazy $\mathcal{AMS}\text{-}\text{\TeX}$ u, se dá očekávat, že tyto potíže budou ještě větší. Skutečně je tomu tak a pravda je bohužel taková, že používat $\mathcal{AMS}\text{-}\text{\LaTeX}$ s běžnými verzemi \TeX u pro PC (což je u nás stále asi nejběžnější) může být poněkud obtížné. Každopádně se většinou budeme pohybovat na pokraji omezení paměti. Potíže bude především činit hlavní paměť.

U běžných implementací \TeX u bývá hlavní paměť omezena číslem přibližně 65 500. V uživatelské příručce $\mathcal{AMS}\text{-}\text{\LaTeX}$ u jsou uvedeny např. tyto údaje, které se týkají zpracování nepřiliš dlouhého článku, který obsahuje asi 50 vlastních definicí příkazů a asi 50 návěstí pro křížové reference. Je-li zpracován standardním \LaTeX em při použití stylu `article`, je využití paměti 51 376 (tyto údaje bývají na konci `.log` souboru). Je-li zpracován v $\mathcal{AMS}\text{-}\text{\LaTeX}$ u ve stylu `article` a voláme-li navíc soubory `amsfonts.sty` a `ambsy.sty`, je kupodivu využití paměti o něco menší — 51 059. Pokud v $\mathcal{AMS}\text{-}\text{\LaTeX}$ u použijeme styl `article` a navíc volbu `amstex`, bude využití hlavní paměti již 63 506. Pokud dále necháme zpracovat tentýž článek ve stylu `amsart` (potom je automaticky volán `amstex.sty` a `theorem.sty`), bude využití hlavní paměti 65 445, což je již skutečně velice blízko maxima.

Zdá se, že používání $\mathcal{AMS}\text{-}\text{\LaTeX}$ u vyžaduje větší implementace \TeX u. Neměl jsem bohužel zatím možnost vyzkoušet $\mathcal{AMS}\text{-}\text{\LaTeX}$ např. s $\text{Big}\text{\TeX}$ em (který je k dispozici v rámci $\text{em}\text{\TeX}$ u) nebo s nějakou jinou větší verzí \TeX u.

(Miroslav Dont)

e-mail: `dont@cseara.bitnet`

Počítačový fond češtiny

Úvod

V tomto textu bychom rádi seznámili členy ζTUG s projektem Počítačového fondu češtiny, který si klade za cíl počítačové zpracování české slovní zásoby a vytvoření rozsáhlé lexikální databáze češtiny, jež bude sloužit jako východisko pro tvorbu českých slovníků všeho druhu.