



bulletin

Československé sdružení uživatelů $\text{T}_{\text{E}}\text{X}_{\text{u}}$
CS TUG

OBSAH

J. Veselý: Čtvrté ohlédnutí ...	1
P. Sekanina: Námět k zamyšlení pro T _E Xovskou komunitu	5
D. Trunec: Ještě jednou o obrázcích v T _E Xu	6
L. Lhotka: České dělení pro T _E X	8
J. Chlebíková: Ako rozdělit (slovo) Československo	10
M. Dont: $\mathcal{A}\mathcal{M}\mathcal{S}$ -L ^A T _E X verze 1.0 a 1.1	13
K. Pala: Počítačový fond češtiny	19
P. Sekanina: MusicT _E X—Sazba notových partů pomocí T _E Xu	25
O. Ulrych: Konkursní řízení	28
L. Durst: Vytváření rejstříků: Část 1	30
O. Ulrych: Kolik časopisů je vydáváno T _E Xem v Československu?	37
O. Ulrych: Redakční poznámka	39

$\mathcal{A}\mathcal{M}\mathcal{S}$ - $\mathcal{L}\mathcal{A}\mathcal{T}\mathcal{E}\mathcal{X}$ akaa K

$\mathcal{A}\mathcal{M}\mathcal{S}$ - $\mathcal{L}\mathcal{A}\mathcal{T}\mathcal{E}\mathcal{X}$ akaa K

Čtvrté ohlédnutí ...

Blíží se konec roku 1991 — prvního roku s $\mathcal{G}\mathcal{T}\mathcal{U}\mathcal{G}$ em, $\mathcal{T}\mathcal{E}\mathcal{X}$ bulletinem a vším tím, co s nimi souvisí. Sedmáctý listopad (den, kdy tento „úvodník“ začínám psát) oslavuji prací pro $\mathcal{T}\mathcal{E}\mathcal{X}$, neboť s ohledem na vývoj situace kolem projektů transformace české vzdělávací soustavy mi dříve čas k napsání úvodníku nezbyl.

Vracím se nejprve k některým problémům: na poslední schůzi se výbor $\mathcal{G}\mathcal{T}\mathcal{U}\mathcal{G}$ u rozhodl zakoupit pro $\mathcal{G}\mathcal{T}\mathcal{U}\mathcal{G}$ korektory českého a slovenského textu od skupiny J. Drózda (dalšími autory jsou J. Hajič a J. Hric). Jejich cena je 48 000 Kčs (za oba korektory). Zároveň se rozhodl odškodnit dr. Pavla Ševečka, který pracoval na úpravě $\mathcal{G}\mathcal{S}\mathcal{E}\mathcal{D}$ u pro $\mathcal{G}\mathcal{T}\mathcal{U}\mathcal{G}$ a nabízel podobné korektory pro češtinu, slovenštinu a angličtinu včetně upravené verze $\mathcal{G}\mathcal{S}\mathcal{E}\mathcal{D}$ u za 75 000 Kčs, nabídnutím uveřejnění stránkové reklamy ve všech číslech příštího ročníku $\mathcal{T}\mathcal{E}\mathcal{X}$ bulletinu nebo vyplacení jednorázové odměny za úpravu ve výši 5 000 Kčs. Program bude archivován, nebude však šířitelný ani v rámci sdružení. Korektory, na jejichž využívání musíme uzavřít smlouvu s firmou SEZAM, jsou již k dispozici, nicméně je nutné je alespoň laicky otestovat (to se právě děje, smlouva však bude umožňovat reklamovat chyby každému z nás; také na úpravu slovníků je pamatováno a první připomínky J. Chlebkové ke slovenské verzi byly zahrnuty již do verze, která bude šířena). Podotýkám ale výslovně, že jde o software, který bude majetkem $\mathcal{G}\mathcal{T}\mathcal{U}\mathcal{G}$ u a prosím všechny členy, aby ho nečlenům za žádných okolností neposkytovali. Smlouvou bude takové jednání jasně vymezeno jako protiprávní! Text smlouvy i manuál k použití dostanete v elektronické podobě.

Zakupované korektory J. Drózda fungují nezávisle na editoru, korektory P. Ševečka musí být vázány na editor $\mathcal{G}\mathcal{S}\mathcal{E}\mathcal{D}$. Jistě pochopíte, že rozhodování výboru nebylo jednoduché a že jen budoucnost prověří, zda jsme vaším jménem rozhodli správně. I v tomto faktoru (jednom z rozhodujících) jsme se nemohli shodnout: co někteří z nás považují za výhodu, jiní hodnotí jako nedostatek.

Osobně mne mrzí, že jsme se jako výbor rozhodovali tak dlouho a neefektivně. Neměli jsme v této oblasti zkušenosti a koncepce výběru na základě připomínek členů výboru se neukázala šťastnou - měli jsme nákup řešit jinak, za to vše však nesu zodpovědnost já.

Dospěli jsme dále k rozhodnutí, že editor koupíme na základě konkursu. Jsou minimálně dva kandidáti: $\mathcal{G}\mathcal{S}\mathcal{E}\mathcal{D}$ a editor Hušek & kol. (jde o syna Mirka Huška, člena výboru $\mathcal{G}\mathcal{T}\mathcal{U}\mathcal{G}$ u a jeho kolegu — kol. tedy nereprezentuje kolektiv, ale kolegu, kterému se tímto omlouvám, neboť jeho jméno neznám), nechceme však propást i případné další možnosti. Na Valném shromáždění 14.12.1991 zvolíme i několikačlennou (cca 3 lidé) komisi, která nabídnuté editory vyhodnotí. Na základě zprávy komise výbor o koupi posléze rozhodne, ale vlastní hodnocení bude provádět komise složená asi ze tří členů, které si zvolíte.

Podmínky konkursu přinášíme uvnitř tohoto T_EXbulletinu; prosím, upozorněte všechny potencionální účastníky, kteří by mohli „T_EX-user-friendly“ editor nabídnout. K nákupu editoru se budete moci během shromáždění vyjádřit. Je zde totiž celá řada public domain editorů (viz článek M. Donta), které lze zařadit okamžitě mezi software, který v rámci sdružení šíříme. Je samozřejmě dostupný i prostřednictvím sítě (editory byly vyhledány v elektronických archívech). Přibyl i editor od DANTE, je k dispozici i demo (= funkční a téměř úplná) verze editoru ME. V té souvislosti bych rád poděkoval všem, kteří nás na existenci zajímavých editorů upozorňovali. Myslím, že k nákupu T_EXově orientovaného a s češtinou a slovenštinou dobře kooperujícího editoru patrně dojde v krátké době (pokud se nezmění prudce ekonomické podmínky sdružení), výbor však bude rád, vyjádříte-li při shromáždění svá přání i celkové své mínění. Je známo, že nejlepší editor je z osobního hlediska jednotlivce ten, se kterým je zvyklý pracovat, proto je dosažení shody značně obtížné.

Dalším problémem, kterým se výbor na své poslední schůzi zabýval, bylo uspořádání konference EuroT_EX 92 v Praze. Vývoj lze shrnout takto: T_EXu bylo u nás věnováno několik seminářů (Cikháj, Skalský Dvůr, . . .), některé z nich pracují v centrech i dlouhodoběji. T_EX je sice znám jen převážně v akademických kruzích, ale jeho obliba, navzdory skepsi některých zastánců práce s jinými programy, stále roste.

Příznávám, že v kuloárech jsme se již dříve zabývali myšlenkou uspořádat v budoucnu větší T_EXovou akci v ČSFR, avšak přeci jen poněkud později. Hodnotíme-li současnou situaci v ČSFR, jeví se nám hned z několika stránek jako ne zcela stabilizovaná a po ekonomické (a státoprávní) stránce ne zcela příznivá pro uspořádání takové náročné akce. Žádný závazek uspořádat větší T_EXovou konferenci v ČSFR příští rok jsme předtím nikdy nepřijali.

Úspěšnost konference ve Skalském Dvoře (Dr. Jaroslav Nadrchal a jeho osvědčený kolektiv), příznivé reference Ericha Neuwirtha o jeho pražské květnové přednášce, růst organizace ζ TUG, časopis T_EXbulletin, šíření národních implementací a získaného softwaru, úspěch Jiřího Zlatušky v Paříži na EuroT_EXu 91 (jeho příspěvek „Automatic generation of virtual fonts with accented letters for T_EX“ získal první cenu; gratulujeme!) a snad ještě další faktory způsobily, že se **někteří** evropští T_EXoví kapitáni rozhodli dát ζ TUGu šanci ukázat, co umí. V Paříži bylo ohlášeno, že příští akce („Seventh European T_EX Conference“) se bude konat v roce 1992 v Praze.

Myslím, že i k této věci přistupoval výbor ζ TUGu s maximální odpovědností. Časový prostor byl značně zúžen termíny pro organizaci takového podniku v Praze byly již dříve rozebrány. Malcolm Clark nás upřímně upozornil, že vzhled, n k pořádání konference v Portlandu (USA) nám jako prezident TUG u pro příští rok **nemůže** slíbit finanční podporu. (Současná finanční situace TUGu není taková aby si TUG mohl dovolit podporovat dvě konference.) Rozhodnutí pokusit se konferenci zorganizovat, jsme přijali zejména proto, že doufáme, že to bude pro ζ TUG i T_EX dobré. Mnohé z nás toto rozhodnutí již stálo a stojí dosti sil a času. Používám této příležitosti, abych všem, kteří neváhali okamžitě přiložit ruku k dílu, touto cestou poděkoval. S tím však souvisí i toto: na Valném shromáždění 14.12.1991 provedeme **doplňující volby** do výboru ζ TUGu s omezenou rotací (to neznamena, že by se výbor nemohl zcela vyměnit, jeho, členové však cítí, že s ohledem na EuroT_EX by nebylo zcela korektní následníky do něčeho namočít a pak od toho utéci). Samozřejmě, že každá ruka (a taky hlava!), přiložená k dílu na této akci, je vítána. Programovou část konference má na starosti Jiří Zlatuška a jeho brněňští kolegové, ostatní pražští členové výboru

a další lidé kolem výboru zajistí organizační záležitosti. Funkci předsedy lokálního organizačního výboru pro tuto akci se uvolil na moji prosbu na svá bedra vzít Karel Horák, který v příštím roce bude také pečovat o \TeX bulletin. Jsem Karlovi za tuto pomoc zavázán a vynasazím se mu pomáhat ze všech sil, je však nutné nadále stále pečovat o ζ TUG i o Euro \TeX a ne vždy je to jednoduché.

Jisté zpoždění, zaviněné nedostatkem příspěvků, mi umožňuje dát příklad a vsunout informaci o tom, jak jsme s Láďou Lhotkou v Klíčanech vysvětlovali při vydávání daru od vydavatelství Addison-Wesley Bonn (jde, mimochodem, o instituci téměř nezávislou na stejnojmenném americkém vydavatelství), co je to \TeX a komu darované knihy jsou určeny. Celní poplatky nám byly prominuty na základě předloženého potvrzení, bahnem jsme se šťastně přebrodili a knihy dovezli. Bilance: ztracené dopoledne, Láďa obětavě ochránil moji bederní páteř (ještě jednou díky) a zmeškal schůzi, na kterou mj. přijel, a vy si budete moci o \TeX u přečíst o něco více. Byl bych však rád, a Láďa jistě také kdyby za to někdo z vás napsal něco pro \TeX bulletin. Mimochodem, strašně nám na MFF UK záleželo z provozních důvodů na termínu uzávěrky 20.11. Chtěli jsme vám zajímavého čtení před Vánocemi poskytnout ještě více, ale nějak to tentokrát nevychází; opět jsme museli vyslat S.O.S. k přátelům, kteří nám již mnohokrát pomohli. Díky nim snad číslo dokončíme, rezervy však žádné (kromě poměrně starých materiálů z nerealizovaného cikhájského sborníku) nemáme. Obsah TUGboatu dnes nehledejte, zatím není ani v USA k dispozici. Měli jsme slíbené i článek o podrobnějším pohledu na sazbu chemických textů, ale potenciální autor na nás pozapomněl (chápu: všichni máme ještě jiné starosti, ale ...). A ještě jeden poznatek: s přepravou **binárních** souborů po **komutovaných** pražských linkách máme prozatím mizerné zkušenosti. Také přechody ze sítě do sítě vedou někdy k obtížně opravitelným chybám. S pomocí Mírka Donta se mnohé podařilo napravit (stažením přímo v OVC), ne ale vše.

Proběhla jednání s Michaelem Spivakem o $\mathcal{L}_A\mathcal{M}_S$ - \TeX u. Mirek Dont přislíbil provést překlad a celá věc vázne jen na podpisu smlouvy, v níž jsou k naší tíži (jedna ze smluvních podmínek, na níž M.S. velice záleží) započteny americké daně (a 10 % z ceny každého výtisku). I zde bude pro nás důležitý váš názor a oceníme i kokoholi, kdo nám pomůže radou s problémem amerického zdanění. V dnešním \TeX bulletinu naleznete další zajímavý příspěvek od M. Donta, který se k tomu volně váže: připomenu základní větu ze Spivakova materiálu „LAMSTEX není AMSLATEX“.

Valné shromáždění ζ TUGu proběhne, jak již bylo řečeno, 14.12.1991 v Modré posluchárně RUK (s ohledem na přání mimopražských se koná v sobotu páteřní doprava je velmi náročná!). Budou mu předcházet od 9.30 hod. dvě přednášky. Jednu (spíše elementární) připravuje **Petr Sojka** z Brna a půjde o **(La)TeX** v příkladech, druhou prosloví náš host a přítel z Francie **Yannis Haralambous**. Nezmění-li se titul, máme se na co těšit, bude přednášet o tom, **Jak \TeX dobyl minulost**. Kdo měl možnost ho slyšet přednášet, jistě rád potvrdí, že je to špičkový \TeX pert, který vytvořil v krátké době špičkové fonty pro sazbu ve starých a často i krajně exotických jazycích (Chcete taky vědět jak? Přijďte se podívat a zeptejte se!). Upozorněte, prosím, orientalisty, teology, odborníky na jazyky, nepsané latinkou, představitele židovských náboženských obcí (ano, \TeX zvládá úspěšně moderní i staré písmo mnoha národů) i odborníky na arabštinu ... Jak vidíte, \TeX spojuje často i nespojitelné.

V tomto \TeX bulletinu najdete i článek o počítačovém fondu češtiny od Karla Paly. I když souvisí s náplní činnosti ζ TUGu jen vzdáleně, uveřejňujeme ho v ne-

zkrácené podobě dobré věci se přeci musí vždy pomáhat. Bude-li se projekt týkat podstatněji i členů ζ TUGu, ukáže zase teprve budoucnost. Podaří-li se plány realizovat, pak snad nebude obtížné zlepšovat korektory, dělení apod.

V minulém čísle se nám sice nakonec podařilo vytisknout tabulku s popisem archivu DANTE (opět díky Láďovi Lhotkovi). Nám — ne jemu — se podařilo pokazit jeho záměr s elegantním očíslováním. Ještě jednou se mu touto cestou omlouvám a omlouvám se i vám. Když jsme na to přišli, bylo již pozdě: $\text{T}_{\text{E}}\text{X}$ bulletin byl už vytištěn. Popis obsahu disket je zcela v pořádku a to je důležité; správné číslování se s trochou hádání zrekonstruovat dá, ale tento rébus jsme vám nepřipravili záměrně.

Možná, že si kladete otázku, co by se stalo s ζ TUGem, kdyby se federace rozpadla Věřím, že nic. Stále by pracoval stejně a snad by ani nemusil měnit název — věřím, že nakonec většina jeho členů by dala přednost unitárnímu státu, který by byl postaven na **občanských** principech, s jednou (ale moudrou) vládou a spravedlivými nikoho nediskriminujícími principy. Obory, kterými se většina z nás zabírá, nabádají spíše k toleranci, nejsme tedy patrně reprezentativní vzorek; rád bych ale připomenul, že za zdůrazňováním toho, co nás dělí, je mnoho zájmů ryze osobních a to je škoda. $\text{T}_{\text{E}}\text{X}$ je nepolitický a tohle rozhodně nemá být politika, působte však, prosím, na svoje okolí.

Dovolu mi pochlubit se cizím peřím: kdysi jsem měl možnost povečeřet u stolu se známým brněnským básníkem, který mi na účtenku napsal na památku „Jsem hotov chodit bos když nechám botu v řiti blbce!“. Tolerance se opravdu hodí jen k tolerantním partnerům. O.M. na závěr večera povzbudil mého leningradského přítele, velice slušného člověka s vyznačeným neruským a neárijským původem v pase (a ten to věru potřeboval, psal se rok 1969 a pohledy, jimiž ho všichni sledovali, nebyly přátelské, i když mluvil a dosud mluví velmi dobře česky), slovy: „Víš, lidi a národy mají k sobě blízko, jen ty vlády stojí za . . .“. Bylo by však škoda, kdyby nás spojoval jen kritický postoj k jednání našich vlád, konec konců nás spojuje alespoň $\text{T}_{\text{E}}\text{X}$; byl bych rád, aby nás však spojovalo o trochu více.

Byl bych na druhé straně velice nerad, kdyby historie ζ TUGu jako společné organizace rozumně myslících Slováků a Čechů (i já si zvykám na to, že je alespoň někdy nutno porušit abecední pořádek, aby naši přátelé věděli, jak moc na ně v těchto dnech v „pragocentrické Praze“ myslím) končila příliš brzo. Konec konců je představitelné, že nejen odlišné jazykové skupiny, nejen národy, nejen země a velká města, ale doslova každý jednotlivec bude mít svoji vlastní verzi $\text{T}_{\text{E}}\text{X}$ u s odlišnými řídicími slovy, odlišnými drivery, odlišnými kódovacími tabulkami, zkrátka vším. V této zemi v poslední době občas slyším argumentaci „Proč by to tak nemohlo být, když je to možné?“. Nemyslím, že by pouhá možnost měla nahradit rozum a cit, spíš jen věřím, že lidi kolem $\text{T}_{\text{E}}\text{X}$ u ani v těžkých dobách zdravý rozum a cit pro to, co je dobré, neztratí.

(Jiří Veselý)

e-mail: ummjv@csearn

Pozn: Považuji za vhodné zdůraznit, že zejména v závěru vyjadřuji pouze své osobní stanovisko, které se nemusí shodovat s názory výboru ζ TUG. -jv-

Námět k zamyšlení pro T_EXovskou komunitu

Nevím, zda jste si všimli, jak se rozrůstá a zvětšuje oblast, ve které T_EX získává pozice a dokazuje svoji všestrannost. S tím souvisí i velký rozvoj specializovaných balíků maker na jedné straně i potřeba kvalitní a dostupné literatury.

Pokud naleznete kdekoliv nějakou pěknou T_EXovou pomůcku a vyzkoušíte ji – napište o svých, byť i prvních zkušenostech s takovým programem. Usnadníte tím velice orientaci dalším našim členům a ušetříme si navzájem čas. Vždyť co vyzkoušel jeden, nemusí pak znovu objevovat druhý z nás, to už nemluvíme o zbytečné dvoj-kolejnosti při vývoji vlastního software. Tady by pomohlo i oznámení úmyslu pustit se do daného problému. Další lidé s podobnými úmysly se pak s vámi mohou spojit přímo.

Návrh na recenze bych měl: L^AT_EXovské prostředí MultiColumn, T_EXCAD pro prostředí picture v L^AT_EXu, L^AM_S-T_EX případně A_MS-L^AT_EX. Další oblasti jsou „překladače“ z jiných textových formátů (z WordPerfectu, z ChiWriteru . . .). Vyzkoušet, odhalit přednosti a slabosti, odhadnout použitelnost, napsat do T_EX–Bulletinu.

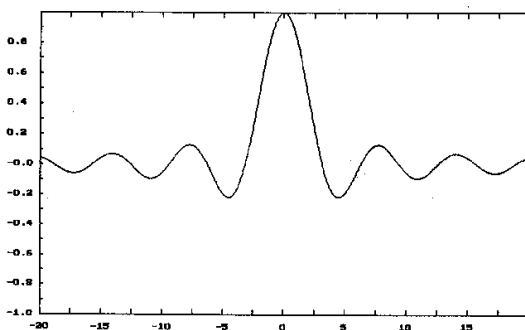
Druhá je otázka literatury. Pokud jste pro vlastní pracoviště vyrobili nějakou příručku o T_EXu nebo se snažíte překládat něco ze zahraniční literatury – dejte vědět na „ústředí“, znova se tím zabrání zbytečnému plýtvání silami. U nás v Brně například je už téměř hotov elektronický překlad Referenční příručky L^AT_EXu. V informaci o literatuře je také zajímavé, kde, kdo a za kolik danou literaturu poskytuje. Nebylo by až tak špatné, kdyby pro členy T_EXovské komunity byla naše „dílnka“ v elektronické podobě „TUG–domain“.

(Pavel Sekanina)

Ještě jednou o obrázcích v \TeX

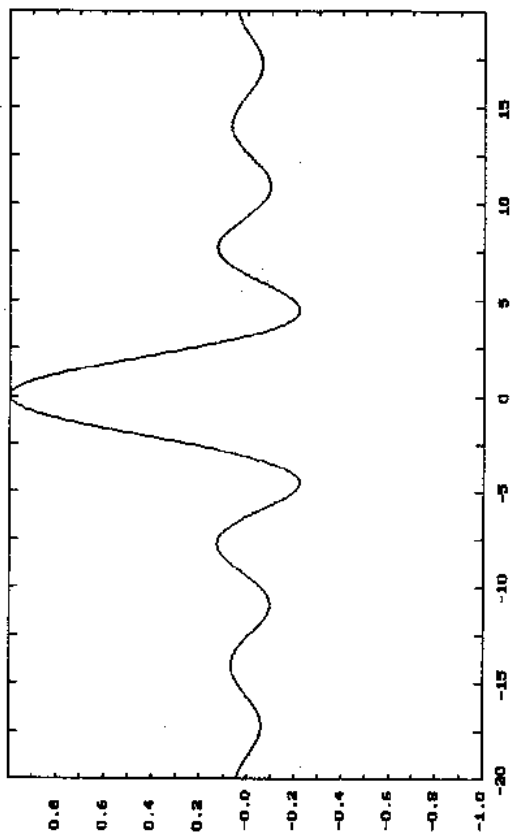
S příchodem $\text{em}\text{\TeX}$ byl částečně vyřešen problém s obrázky v \TeX . Program $\text{\TeX}cad$ umožňoval poměrně pohodlně kreslit obrázky, navíc bylo možno pomocí příkazu $\backslash\text{special}$ začlenit do textu grafiku ve formátu *pcx* nebo *msp*. Jediné, co scházelo, byla možnost začlenit do \TeX hardcopy obrazovky. Jestliže se spočítaly nějaké výsledky, např. programem v Turbo Pascalu, a vykreslily do grafu na obrazovce počítače, pak se muselo udělat hardcopy obrazovky na tiskárně, v textu vysázeném \TeX em vynechat místo a obrázek tam vlepít. Sháněl jsem nějaké programy, které by mi umožnily sejmut obsah obrazovky ve formátech *pcx* nebo *msp*, ale na žádné rozumné jsem nenarazil. Navíc zde byl problém s velikostí obrázků. Jestliže obrázek měl rozměr 640×480 bodů, tak při hustotě 300 dpi, byl výsledný obrázek neúnosně malý.

Nakonec se našlo řešení svépomocí. Kolega dr. Marek napsal program pro hardcopy na laserovou tiskárnu Hewlett Packard IIP, který vlastně generoval sekvence PCL. Pak již bylo snadné tento program modifikovat tak, aby se vytvářel soubor na disku ve tvaru PCL. Tento soubor se pak konvertuje programem PCLtoMSP z $\text{em}\text{\TeX}$ u na formát *pcx* nebo *msp*, který je již možno začlenit do textu. Dále je možno volit velikost obrázku a obrázky jsou dostatečně velké. Tento program se jmenuje CPCL a je to rezidentní program, který v paměti zabere 3 kB. Při spuštění programu je možno volit velikost obrázku pomocí hustoty (75 dpi je největší obrázek, 100 dpi menší atd.). Bohužel výsledná velikost obrázku závisí ještě na použitých fontech pro tisk (při tisku na laserové tiskárně s fonty 300×300 dpi je obrázek menší než při tisku na jehličkové tiskárně s fonty 240×216 dpi). Další vadou na kráse je to, že soubory typu *pcx* vytvořené programem PCLtoMSP nelze prohlížet ani dále zpracovávat, protože chybí odpovídající hlavička souboru. Následující obrázek byl nakreslen pascalovským programem při rozlišení 640×480 bodů a pak byl sejmut s hustotou 150 dpi, druhý obrázek s hustotou 100 dpi.



Funkce sinc (menší obrázek)

Doba sejmutí a uložení jednoho obrázku je asi 50 sekund. Program CPCL je k dispozici u autora, nejjednodušší způsob kontaktu je pomocí EARNu.



Funkce sinc (větší obrázek)

Dále bych chtěl upozornit na balík programů Laser Graphics, který je možno získat ze SIMTELu. Je to celkem 10 souborů v adresáři MSDOS.PLOT, jejichž názvy jsou LG-*.ZIP. Tento systém umožňuje vytvářet nejrůznější grafy (jedno- i vícedimenzionální) a pak vygenerovat příslušné obrázky ve formátu PCX s velkým rozlišením. Lze je tedy pak velmi snadno zabudovat do $\text{T}_{\text{E}}\text{X}$ u. Dobrou ukázkou je samotná dokumentace, která byla napsána v $\text{T}_{\text{E}}\text{X}$ u.

(D. Trunec)

e-mail: TRUNEC@CSPUNI12.EARN

České dělení pro T_EX

Motto: "... patterns are supposed to be prepared by experts who are paid well for their expertise."

(D. Knuth, *The T_EXbook*)

Začátkem měsíce listopadu jsem dokončil některé podstatnější úpravy souboru `cshyphen.tex` obsahujícího vzory pro české dělení. Děkuji tímto Karlu Horákovi, který mi předal poměrně rozsáhlý soubor se slovy, která předchozí verze dělila špatně nebo neúplně. Nová verze vzorů je k dispozici v „ústředí“ ζ TUGu (u Olina Ulrycha), případně je mohu přímo zaslat elektronickou poštou.

Při této příležitosti jsem se znovu rozpomněl na idylické doby vzniku původní amatérské české lokalizace T_EXu. Moje první varianta vypadala tak, že jsem příslušný počet znaků ze začátku ASCII tabulky deklaroval jako aktivní (například písmeno š mělo kód "08) a jejich definice odpovídala standardním plovoucím akcentům typu `\v s`. Na prvních 32 míst se vešla všechna malá písmena s akcenty a z velkých písmen ta, která se mohou vyskytnout v běžném textu¹—ostatní velká písmena, například v nadpisech, se realizovala pomocí `\v` a `\'`. Pro vkládání zdrojového textu na PC a jeho případné tištění na tiskárně ovšem toto rozložení nebylo vhodné, protože některé z inkriminovaných kódů mají řídicí funkce, třeba kód "1B (ESC) zapíná povolený režim většiny tiskáren. Zde byla pomoc snadná: Z hlediska počítače se všechny kódy posunuly 0 128 pozic výše, kde nepůsobily problémy. Sedmibitový T_EX pak nejvyšší bit uřízl a všechno bylo v pořádku. Jediným problémem bylo dělení, protože před verzí 3.0 nedělil T_EX slova obsahující aktivní znaky nebo makra.

Někdy na podzim roku 1988 jsme se domluvili s Petrem Novákem a začali uvažovat o „profesionalizaci“ češtiny (a slovenštiny) v T_EXu (oba jsme tehdy byli aspiranty a měli proto na takové věci čas). Přes značné odhodlání jsem nakonec neudržel své kódování češtiny v PC proti standardu Kamenických, čímž ovšem vyvstala nutnost filtrovat vstupní soubory (tedy to, co emT_EX zařizuje sám pomocí `tcp` tabulek). Vyzjasnili jsme si také, že slušné české dělení je možné jen tehdy, vyrobíme-li T_EXovské fonty obsahující přímo akcentovaná písmena, a dále rozsáhlejší soubor vzorů pro české dělení. Vzhledem k tomu, že jsem měl k dispozici METAFONTbook, dohodli jsme se, že já připravím fonty a Petr se bude věnovat dělení.

Vše nakonec dopadlo zcela jinak. Petr si okopiroval METAFONTbook a, propadnuv jeho nespornému kouzlu, začal experimentovat s písmenky. Takto postupně vznikly jeho CS fonty, které byly první (a jsou zřejmě dosud jedinou) kvalitní československou mutací rodiny *Computer Modern*. No a na mne tedy zbylo dělení.

Půjčil jsem si v knihovně prastarou knihu Jiřího Hallera *Jak dělit slova* (SPN 1956) a začal studovat. První kroky byly snadné. Základní vzory dělení v češtině i slovenštině jsou typu, *samohláska-souhláska*, *samohláska a samohláska*, *souhláska-souhláska*, *samohláska*. Napsal jsem proto program, který tyto kombinace generoval. Přímou program a pak i ručně jsem eliminoval nemožné skupiny hlásek typu *úvř* (doufám, že v tomto článku nebude nutno tuto skupinu dělit). Nejobtížnější částí bylo zpracování předpon, kde se možnosti dělení zpravidla rozsáhle větví (*pře-*, *před-*, *přede-*). Musel jsem projít celý (nepříliš rozsáhlý) slovník v Hallerově knize a uvázat

¹ nikoliv tedy např. ě

všechny možné varianty a výjimky. Podobně to bylo s příponami. Nakonec přišly na řadu cizí předpony a přípony a izolované výjimky. Přibližně po dvou týdnech usilovné práce byla první verze hotova.

Počáteční fáze testování a úprav ukázala některé důležité věci:

1. Ve spojení s \TeX em umožňuje tato tabulka dělení docela uspokojivě sazbu českého textu bez nutnosti příliš častých zásahů
2. Protože je soubor uspořádán do logických celků, je poměrně snadné doplňovat postupně další vzory a uvážit všechny možné vedlejší efekty.
3. Největší problémy jsou ve slovech složených anebo obsahujících více předpon (například *nejneobvyklejší*). Tato negativní vlastnost v podstatě přetrvává i ve všech dalších verzích. Naštěstí nemáme v našich jazycích takovou frekvenci složenin jako třeba němčina (*Donaudampfschiffsfahrtkapitänkajütentürschlo*).

I přes původní záměr jsem nakonec rezignoval na současné zpracování slovenské části vzorů. Správně jsem předpokládal, že se tohoto úkolu zhostí dříve či později někdo ze Slovenska (učinila tak Janka Chlebíková).

S výsledkem jsem byl s ohledem na vložené úsilí velice spokojen. Ukázalo se, že v mnoha případech automatické dělení překonalo svého tvůrce, když jsem ke svému překvapení musel po prověření ve slovníku opravit své mylné představy. Postupně se sice objevovala některá více či méně fatální opomenutí (kupříkladu *Česko-slovensko*—slovenští přátelé, neříkejte to na mě vašim pomlčkovým mamelukům), která jsem byl, většinou bez větší námahy, schopen průběžně doplňovat.

Větší problémy mi paradoxně přinesl až \TeX 3.0, který mimo jiné odstranil tvrdé pravidlo předchozích verzí, že totiž ze slova lze oddělit nejméně dvě písmena na začátku a tři na konci. S tímto omezením jsem mlčky počítal při přípravě vzorů. Po nastavení parametrů `\lefthyphenmin` a `\righthyphenmin` na místním zvyklostem odpovídající hodnoty 1 a 2 jsem s hrůzou zjistil, že se dělí třeba *h-růza*. Proto bylo třeba podniknout rozsáhlejší revizi vzorů.

V poslední verzi jsem zejména doplnil některé časté dvojice předpon (nepřepod.), další cizí předpony, složené číslovky a jiná důležitá složená slova. Karel Horák mi doufám odpustí, že případy jako *SNTL*, *JČMF* či *Tartaglia* si bude muset nadále ošetřovat sám.

Je zřejmé, že ani poslední verze není dokonalá. Pokud tedy naleznete případy chybného dělení či nedělení, podejte mi prosím zprávu. Rychlost odezvy ovšem nemohu garantovat, nejsem už totiž aspirant. Výhledově by bylo pravděpodobně užitečné připravit w nejuplněnější soubor vzorů pomocí speciálního programu PATGEN na základě nějakého velkého slovníku. Petr Novák mezitím něco takového udělal, svůj výsledek však zatím zřejmě neposkytuje volně \TeX ové komunitě. Momentálně se tedy musíme smířit s občasnou ruční opravou případů, kde můj skromný (zato ale *public domain*) příspěvek selhává. Není vyloučeno, že jím nepohrdnou ani Lidové noviny—myslím si, že by to bylo každopádně lepší než jejich současné dělení.

(Ladislav Lhotka)

e-mail: lhotka@csearn

Ako rozdeliť (slovo) Československo.

Každý, kto používal \TeX na písanie v slovenčine či češtine v dobe, kedy \TeX ešte nebol prispôbený na naše československé (česko-slovenské, české a slovenské)² pomery, musel sa vysporiadať s dvoma nákladnými problémami:

1. Akcentované znaky – špeciálne I , d , t , ($\backslash\text{v t}$ totiž láva t , s čím sa nemožno uspokojiť). Nemožno tiež zabudnúť na „naše“ úvodzovky, ktoré sa líšia od „anglických“.
2. Rozdeľovanie slov – neprispôbené pôvodné verzie \TeX u rozdeľujú slová podľa pravidiel anglickej gramatiky, vôbec nerozdeľujú akcentované slová a ani ich nedovoľujú pridávať do slovníka výnimiek $\backslash\text{hyphenation}\{\dots\}$.

Prvý problém je možno v súčasnosti veľmi pohodlne riešiť (odhliadnuc od pamäťových nárokov) napr. použitím 8-bitových fontov od P. Nováka. Pred dokončením sú už i 8-bitové DC-fonty, obsahujúce väčšinu znakov z najrozšírenejších európskych jazykov (i naše t , d , I , f). Tieto fonty by sa mali stať medzinárodným štandardom. Navyše všeobecne rozšírený emTeX poskytuje pohodlnú možnosť konverzie akcentovaných znakov do ľubovoľných kódov, takže editovanie si môžeme spríjemniť napr. použitím známeho programu od Kamenických. Podľa mojich osobných skúseností ani posledná verzia PC \TeX u neumožňuje tak jednoduchú konverziu ako emTeX (Američanov až natalko netrápia naše akcentované znaky).

S druhým problémom sa musí každý jazyk vysporiadať sám vytvorením vlastného „vzorkovníka“. Pre češtinu ho vytvoril Láda Lhotka a české delenie funguje celkom úspešne. I napriek tomu, že naše jazyky sú blízke, gramatické jemnôstky ma viedli k pokusom s vytváraním slovenských vzorov pre delenie (a nie poslovenčením českých, čo by sa na prvý pohľad mohlo zdať jednoduchšie).

Rozdeľovanie slov na počítači sa nedá urobiť dokonale. Správnosť rozdelenia často závisí od kontextu, napr. na-drobiť a nad-robiť. Autorom rozdeľovacieho algoritmu pre \TeX , ako i „vzorkovníka“ pre angličtinu je Frank M. Liang. Svoje výsledky (hlavne problém vytváranie vzorov) zhrnul do Ph. D. Thesis (1983) na Stanfordovej Univerzite. Navrhnutá metóda je dostatočne rýchla a láva možnosť pomocou iných vzorov učiť \TeX rozdeľovať i v iných jazykoch (napr. v slovenčine alebo v češtine). Kvalita delenia závisí potom už len od nás, koľko času venujeme príprave vzorov.

\TeX sa pri delení slova najskôr pozrie do slovníka výnimiek $\backslash\text{hyphenation}\{\dots\}$. Ak tam slovo nájde, rozdelí ho spôsobom, ako mu predpisujeme, napr. $\backslash\text{hyphenation}\{\text{výnim-ka}\}$. V takomto prípade už nepoužíva delenie pomocou „vzorkovníka“ (takže uvedením slova do bez rozdeľovníkov možno zabrániť rozdeľovaniu slova. v celom dokumente). Obdobne je tomu i u slov, v ktorých sa zadáva delenie explicitne pomocou $\backslash-$, napr. $\text{po}\backslash\text{-mlč}\backslash\text{-ka}$.

Pokiaľ nenastane ani jedna zo spomínaných možností, spustí Liangov algoritmus. Ukážme si, ako bude rozdelené slovo **Czechoslovakia** s anglickými vzormi zo súboru *enhyph.tex* (štandardne distribuovaného s $\zeta\TeX$ om). Pri rozdeľovaní slov veľké a malé písmen nehrajú rolu, preto \TeX najskôr „preloží“ slovo do malých písmen (samozrejme tento preklad musí byť presne definovaný). Na začiatok a koniec slova pridá extra znaky – bodky, ktoré budú v ďalšom signalizovať, že sa jedná o špeciálnu časť slova (začiatok a koniec).

² Každý nech si vyberie podľa svojho zmýšľania, či politickej príslušnosti.

Označované slovo

.czechoslovakia.

(dĺžky 16) rozdelí na všetky podslová dĺžky 1, 2, ... Takto dostáva nasledujúce podslová:

dĺžky 1: ., c, z, e, ...

dĺžky 2: .c, cz, ze, ...

dĺžky 3: .cz, zec, ech, ...

...

dĺžky 15: .czechoslovakia, czechoslovakia.

dĺžky 16: .czechoslovakia.

Potom pre každé obdržané podslovo nájde medzi vzormi jeho „ohodnotený“ (celými čísla od 0 do 9) ekvivalent (nemusi vždy existovať). V našom prípade nájde v spomínanom súbore nasledujúce „ohodnotené“ podslová:

cz₄, 2ze, 2ch, 3cho₂, os₄l, 4lov, 1va

podľa ktorých priradí hodnotu každému znaku v rozdeľovanom slove spôsobom:

- znaku, za ktorým nenasleduje žiadne číslo – priradí 0,
- znaku, za ktorým sa stretne viac čísel – priradí najväčšie z nich.

Chceme napríklad určiť hodnotu znaku e v spomínanom slove. Podľa vzorky 2ch tam prislúcha 2 podľa 3cho₂ hodnota 3, ale podľa cze₄ hodnota 4. Výsledná hodnota znaku e je teda 4. Uvedeným spôsobom určíme hodnoty všetkých znakov, čím dostaneme (0 nepíšeme)

cz₄e₄cho₂s₄lo₁v₁vakia.

A *prekvapivý záver*: T_EX rozdelí slovo len za znakmi, ktoré majú nepárnu hodnotu. V našom prípade deliace miesto je teda jediné

czechoslo-vakia.

Pritom sa ešte kontrolujú momentálne nastavené hodnoty `\lefthyphenmin` (resp. `\righthyphenmin`) udávajúce koľko prvých (resp. posledných) znakov nesmie byť v slove rozdelených: pre angličtinu býva `\lefthyphenmin=2` a `\righthyphenmin=3`. (Toto sa ovšem nevzťahuje na slová rozdeľované pomocou `\hyphenation{...}`.)

Presvedčiť sa, že T_EX skutočne rozdelí uvedené slovo daným spôsobom môžeme napísaním `\showhyphens{Czechoslovakia}` do svojho dokumentu a možné rozdelenia slova budú zapísané do súboru LOG. (Možno použiť i na celý paragraf.)

Viac informácií o niektorých jemnosťkach pri rozdeľovaní slov v T_EXu môžu zvedavejší čitatelia nájsť v „biblii“ T_EXbook [1, str. 449–455. Nájdete tam tiež poznámku o tom, že vytváranie „vzorkovníkov“ je dobre platená práca pre expertov. ...

Napriek tomu, že medzi spomínaných expertov nepatrim, každodenné „T_EX-ovanie“ ma privedlo k nutnosti vytvoriť aspoň nejakú verziu slovenských vzorov pre delenie. Na rozdiel od Lianga, ktorý základ svojich vzorov vyrobil z obrovského množstva slov a ich odvolení (k dispozícii mal Webster's Pocket Dictionary v elektronickej forme – zhruba 50 000 slov), použitá metóda je založená priamo na prepise gramatických pravidiel na rozdeľovanie slov. Možno niekomu poslúži ako inšpirácia na vytvorenie dokonalejšej verzie (prípadne poskytnite mi nápady na vylepšenie), alebo ako ukážka toho, kadiaľ cesta nevedie.

Prvá fáza spočíva v prepise mechanických pravidiel rozdeľovania (t.j. pravidiel typu – ak je medzi dvoma samohláskami jedna spoluhláska, ...). Tieto dávajú nie vždy uspokojivé výsledky pri príponách a predponách, čo však je len chybou krásy a

nie gramatickou! V [2, str. 53] sa totiž píše, že ak si rozhranie medzi predponou, slovným základom a príponou neuvedomujeme, pripúšťa sa i mechanické rozdeľovanie slov. (Teda obe rozdelenia „náj-dem“ i „ná-jdem“ sú správne.) V prípade mechanického rozdeľovania slov vznikajú problémy so zloženými slovami, napr. rozdelenie Československo je hrubou (gramatickou) chybou.

Pre slovenčinu obe hodnoty $\backslash\text{lefthyphenmin}$ a $\backslash\text{righthyphenmin}$ nastavíme na 2. Podlá [2] totiž pri rozdeľovaní neoddeľujeme obyčajne koncovú a začiatočnú slabiku, ak obsahuje iba samohlásku.

Na začiatku zabránime deleniu dvojhlasok i_2e , i_2u , i_2a a spoluhlások d_2z , $d_2ž$, c_2h . Potom pridáme každú samohlásku samostatne s 1:

a_1 , l , \ddot{a}_1 , e_1 , \acute{e}_1 , ...

Tým si zabezpečíme správne delenie „pekných“ slov, v ktorých sa pravidelne strieda spoluhláska so samohláskou (slov typu „ko-lá-č“ máme ošetrené s $\backslash\text{righthyphenmin}$). Ďalšie pravidlo: ak sú medzi samohláskami dve spoluhlásky (r, l počítam vždy medzi spoluhlásky) prepíšeme do tvaru:

$z b_1 b$, $z b_1 t$, $z b_1 \check{c}$

pre všetky možné dvojice spoluhlások, vrátane dz , $dž$, ch . *POZOR*: dvojice $z c_1 h$, $z d_1 z$, $z d_1 ž$ musíme zo zrejmých dôvodov vyčiarknuť zo „vzorkovníka“.

Pridaním ďalšieho pravidla – výskyt troch spoluhlások medzi dvoma samohláskami pridáme do slovníka „vhodné“ trojice v ohodnotení:

..., $s_3 t_2 r$ (ses-tra), $n_3 d_2 r$ (han-dra), $n_3 s_2 k$ (pán-sky), ...

Problém výberu takýchto trojíc je zložitejší ako v predchádzajúcom prípade dvojíc, lebo nemôžeme pridať všetky možné trojice (a to nielen kvôli kvantite). Napr. pridaním trojice $t_3 \check{I} k$ by sme pokazili správne rozdelenie slova otlkať, ktoré dostaneme cez dvojice. Problém súvisí s neslabičnými l , \check{I} , r , \check{r} , ktoré v týchto prípadoch plnia funkciu samohlásky. Problém sa teda sústreďuje na získanie zoznamu všetkých možných trojíc spoluhlások, vyskytujúcich sa v slovenských slovách. Nejaké štatistické výsledky možno nájsť v [3], rozhodne však nie sú uspokojivé.

Analogicky sa postupuje pri 4, 5, ... spoluhláskach, vyskytujúcich sa spolu. Problém vyhľadávania a ohodnocovania takýchto „zhlukov“ je podobného charakteru ako v predchádzajúcom prípade.

..., $r_2 s_3 t_2 v$, ...

V druhej fáze sa sústreďujeme na predpony. Doterajší prepis zabraňuje rozdeľovaniu niektorých predpôn, napr. v slove „vystáť“ (po predpone vy-). Slovo bude rozdelené vys-táť. Nasleduje teda pridávanie predpôn

..., $.d_03k_4r$, $.d_03k_4l$, ...

Zdôrazňujem, že uvedenie častí slovného základu (niekedy i viac než jedнопísmenovej), je nutné. Ináč by sme pripustili nesprávne delenia, napr. delenie slova do-ktor.

Súčasťou tejto fázy je i pridávanie najfrekventovanejších častí zložených slov (v podstate čosi ako slovná predpona):

..., 5viac_3h_4 (viac-hlasný), 5česko_5s_4 , $.p_04d_3z$, $.štvo_r_3r$...

i s uvedením častí slovného zkladu z rovnakého dôvodu ako v predšlom.

V tretej fáze sa zameráme na prípony. Ich „pridanie“ do úvah spočíva v úprave ohodnotenia „zhlukov“, prípadne ich presnejšej špeciifikácii v spojení s väčšou časťou slovného základu.

Nakoniec nasleduje pridávanie pravidiel pre cudzie slová

e1a2, u1 2, ...

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

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

Literatura

- [1] Knuth D. E., *The TeXbook*, 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 $\mathcal{L}\mathcal{A}\mathcal{M}\mathcal{S}$ - \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 $\mathcal{L}\mathcal{A}\mathcal{M}\mathcal{S}$ - \TeX a $\mathcal{A}\mathcal{M}\mathcal{S}$ - \LaTeX není jedno a totéž a tyto systémy jsou dokonce nekompatibilní. Systém $\mathcal{L}\mathcal{A}\mathcal{M}\mathcal{S}$ - \TeX je rozšíření $\mathcal{A}\mathcal{M}\mathcal{S}$ - \TeX u, což mimo jiné vychází ze skutečnosti, že autorem $\mathcal{L}\mathcal{A}\mathcal{M}\mathcal{S}$ - \TeX u je M. D. Spivak (tj. autor $\mathcal{A}\mathcal{M}\mathcal{S}$ - \TeX u). $\mathcal{L}\mathcal{A}\mathcal{M}\mathcal{S}$ - \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 $\mathcal{L}\mathcal{A}\mathcal{M}\mathcal{S}$ - \TeX od \LaTeX u nepřebírá, je okolí `picture` (okolí pro tvorbu jednoduchých obrázků přímo ve zdrojovém textu).

Na druhé straně dává $\mathcal{L}\mathcal{A}\mathcal{M}\mathcal{S}$ - \TeX velké možnosti při vytváření komutativních diagramů; kdo v textu potřebuje mnoho komutativních diagramů, tomu bych doporučil $\mathcal{L}\mathcal{A}\mathcal{M}\mathcal{S}$ - \TeX . Další výhodou $\mathcal{L}\mathcal{A}\mathcal{M}\mathcal{S}$ - \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 $\mathcal{L}\mathcal{A}\mathcal{M}\mathcal{S}$ - \TeX).

Na rozdíl od $\mathcal{L}\mathcal{A}\mathcal{M}\mathcal{S}$ - \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 Lamportova manuálu (pouze příkazy pro volbu fontů budou mít poněkud jiný význam jak se zmíním dále). Kdo

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

Balík obsahující $\mathcal{A}\mathcal{M}\mathcal{S}$ - \LaTeX se skládá v podstatě ze dvou částí. Jedna část se skládá z velkého množství \LaTeX -souborů a je vlastní nadstavbou \LaTeX u — o těchto souborech se zmíním dále. Druhá část se týká především způsobu výběru fontů a tvoří pevnou část $\mathcal{A}\mathcal{M}\mathcal{S}$ - \LaTeX u (zatímco nové stylové soubory uživatel může ale nemusí v $\mathcal{A}\mathcal{M}\mathcal{S}$ - \LaTeX u používat, změna schématu volání fontů bude zabudována přímo v novém \LaTeX souboru). Podívejme se nyní stručně na tuto pevnou část $\mathcal{A}\mathcal{M}\mathcal{S}$ - \LaTeX u. 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 \LaTeX u a které by mělo být součástí budoucí nové verze \LaTeX u (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 nedefinuje přístup k těmto fontům - zvláště pro začátečníka to může být nepříjemná záležitost. Ve standardním \LaTeX u 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 — odkazují 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 \LaTeX u. Tak např. použijeme-li nyní po sobě příkazy $\text{\sl}\text{\bf}$ (nebo také v obráceném pořadí), bude dále tištěno tučné nakloněné písmo (opět pokud bude k dispozici příslušný font).

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

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, $\mathcal{A}\mathcal{M}\mathcal{S}$ - \LaTeX umožňuje mimo jiné pracovat s AMS-fonty — možnost práce s AMS-fonty však není důvod proč používat $\mathcal{A}\mathcal{M}\mathcal{S}$ - \LaTeX .

Jak jsem již poznamenal, další část $\mathcal{A}\mathcal{M}\mathcal{S}$ - \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 $\mathcal{A}\mathcal{M}\mathcal{S}$ - \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 $\mathcal{A}\mathcal{M}\mathcal{S}$ - \TeX u verze 2.0. Je třeba říci, že příkazy zde definované nejsou zcela identické s příkazy v $\mathcal{A}\mathcal{M}\mathcal{S}$ - \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 $\mathcal{A}\mathcal{M}\mathcal{S}$ - \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 `\!` (nevidím dost dobře situaci, kdy by bylo skutečně potřeba takto jemné mezerování — ostatně v takovémto případě by bylo možné příslušnou mezeru vyrobít různými jinými způsoby). Stejně jako v $\mathcal{A}\mathcal{M}\mathcal{S}$ - \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 je zvyklý na příkazy `\ldots` a `\cdots` (a dále `\vdots` a `\ddots` pro použití v maticích). V soulase s $\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\text{\TeX}$ 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 $\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\text{\TeX}$ 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 $\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\text{\TeX}$ u. \LaTeX ový příkaz `\stackrel{rel}{}` 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 $\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\text{\TeX}$ u lze generovat řetězové zlomky.

V \LaTeX u se k psaní matic většinou používá okolí `array`. Toto okolí je v $\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\text{\LaTeX}$ u zachováno, ale z $\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\text{\TeX}$ 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 $\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\text{\LaTeX}$ u k dispozici např. okolí

```
align      gather      alignat    multiline
split     xalignat    xxalignat
```

Nebudu zde tato okolí popisovat podrobně. Chceme-li vsunout řádek nebo několik řádků textu mezi řádky takto zarovnané formule, můžeme jednoduše použít příkazu `\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 $\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\text{\LaTeX}$ em dodávána v elektronické podobě. Jelikož se jedná o soubor psaný v \LaTeX u (aby bylo možné příručku vytisknout ještě před instalací $\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\text{\LaTeX}$ 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 $\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\text{\LaTeX}$ 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 $\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\text{\LaTeX}$ em se ještě zmíním dále). Psaní komutativních diagramů je zde stejně jako v $\mathcal{A}\mathcal{M}\mathcal{S}\text{-}\text{\TeX}$ 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}{ccc}
S^{\{\cal W\}_\Lambda} \otimes T & \xrightarrow{j} & T \\
\Big\downarrow & & \Big\downarrow \text{End } P \\
(S \otimes T)/I & \xlongequal{\quad} & (Z \otimes T)/J
\end{array}
\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 @VVV \text{End } P \\
(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 nejrůznějších 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`, `ambsy.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 pokadě 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 `ambsy.sty` obsahuje definice příkazů `\boldsymbol` a `\pmb`. Příkazem `\boldsymbol` lze tisknout tučné matematické symboly (pokud existují příslušné fonty). Např. příkazy

```
\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

```
intlrm.sty nonamelm.sty nosumlim.sty righthtag.sty ctagsplt.sty
```

Použití těchto souborů má následující význam (názvy jsou ostatně samovysvětlující). Při použití souboru `intlrm.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 `righthtag.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 částí 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 všechny texty 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 Schöpfy je u nás asi nejznámější styl `multicolumn`, který umožňuje psaní 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 $\mathcal{A}\mathcal{M}\mathcal{S}$ - \LaTeX u, který kombinuje \LaTeX s mnoha příkazy $\mathcal{A}\mathcal{M}\mathcal{S}$ - \TeX u, se dá očekávat, že tyto potíže budou ještě větší. Skutečně je tomu tak a pravda je bohužel taková, že používat $\mathcal{A}\mathcal{M}\mathcal{S}$ - \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 $\mathcal{A}\mathcal{M}\mathcal{S}$ - \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 $\mathcal{A}\mathcal{M}\mathcal{S}$ - \LaTeX u ve stylu `article` a voláme-li navíc soubory `amsfonts.sty` a `amsbsy.sty`, je kupodivu využití paměti o něco menší — 51 059. Pokud v $\mathcal{A}\mathcal{M}\mathcal{S}$ - \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 $\mathcal{A}\mathcal{M}\mathcal{S}$ - \LaTeX u vyžaduje větší implementace \TeX u. Neměl jsem bohužel zatím možnost vyzkoušet $\mathcal{A}\mathcal{M}\mathcal{S}$ - \LaTeX např. s $\text{Big}\mathcal{T}\mathcal{E}\mathcal{X}$ em (který je k dispozici v rámci `em\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.

I když je to snad všeobecně známo, pokládáme za potřebné připomenout, že významné slovníky hlavních evropských jazyků (angličtiny, francouzštiny, němčiny, italštiny, španělštiny a také ostatních) jsou v současnosti vytvářeny pomocí počítačů a také existují v tzv. *počítačově čitelné podobě* jako tzv. „machine readable dictionaries“. Jako příklady lze uvést Longman Dictionary of Contemporary English (LDOCE), Collins COBUILD English Language Dictionary, Oxford English Dictionary (OED), Oxford Advanced Learner’s Dictionary of Current English (OALDCE), Merriam-Webster Seventh New Collegiate Dictionary a některé další. Počítačově čitelné slovníky slouží pak jako východisko pro tvorbu rozsáhlých *lexikálních databází*, které se stávají zdroji pro různé typy slovníků, např. pro dvojjazyčné či jinak specializované slovníky, jako jsou třeba slovníky pro počítačové zpracování přirozeného jazyka (Natural Language Processing) a umělou inteligenci (Artificial Intelligence).

Předpoklady

Lze konstatovat, že při formulování lexikografického projektu pro češtinu bylo již dosaženo celkové shody v několika základních oblastech, které mohou být zajímavé i pro členy ζ TUG a uživatele systému $\text{T}_{\text{E}}\text{X}$.

1. Standardizace

Standardizace spočívá ve vytvoření co možná nejjednodušších způsobů počítačového zpracování češtiny.

1. Všechny podstatné problémy v oblasti kódů i editorů lze vyřešit připojením k celosvětovému projektu TEI (TEXT ENCODING INITIATIVE) a použitím v něm navržené techniky SGML (STANDARD GENERALIZED MARKUP LANGUAGE) (C. M. Sperberg-McQueen & Lou Burnard, 1990).
2. Pro vlastní práci na projektu však pokládáme za nezbytné, aby výchozím standardem se pro zainteresovaná pracoviště stal kód LATIN II, který navazuje na ostatní národní (západoevropské) kódy a je východiskem pro novou čs. normu. Skutečnost, že se užívá více různých kódů (s výraznou převahou kódu M. a J. Kamenických) lze bez větších obtíží řešit jednotným souborem konverzních programů, jenž by byl k dispozici všem zúčastněným pracovištím a pravděpodobně také členům ζ TUG, pokud o to projeví zájem.
3. standardní techniky zpracování typografických textových souborů (konverzní programy), které budou východiskem pro vytvoření reprezentativního korpusu českých slov (v první fázi cca 5–7 mil. slovních forem, v dalších fázích pak 20 a více mil. slovních tvarů). V tomto bodě se obracíme k členům ζ TUG s výzvou ke spolupráci, která může přinést prospěch všem zúčastněným.

2. Právní vztahy

Český korpus se bude vytvářet zpracováním typografických textových souborů, proto je nezbytné navázat kontakty s nakladatelstvími a tiskárnami a s pomocí právníků formulovat smlouvy, které umožní zpracovávat typografické soubory v souladu s platnými zákony o ochraně autorských práv. Návrh takové smlouvy opírající se o zahraniční zvyklosti je již k dispozici.³

³ Lze konstatovat, že nedávno proběhla úspěšná jednání mezi členy Skupiny a redakcemi LIDOVÝCH NOVIN, MLADÉ FRONTY DNES o poskytování textových souborů vhodných pro vytváření korpusu češtiny. Navazují se kontakty s tiskárnami: Po-

Jádro projektu

Za hlavní cíl českého lexikografického projektu pokládáme vybudování **počítačového fondu češtiny**, který bude tvořen **lexikální databází současné spisovné češtiny a dílčími databázemi** zahrnujícími běžně mluvený jazyk, české a moravské dialekty a také starou a starší češtinu. Pro budování triády tvořené databází současné češtiny a dvěma dalšími databázemi se s výhodou použije jednotného programového vybavení a pokud možno vzájemně kompatibilních metodologických postupů.

Současná spisovná čeština

Pro lexikografické zpracování současného českého jazyka tedy předpokládáme tři základní úrovně:

1. úroveň **datové báze současných českých textů**, která by měla pokrýt slovní zásobu moderní češtiny a sloužit jako základ pro postupně budovaný reprezentativní korpus češtiny čítající kolem 20 miliónů slovních forem (pro srovnání s angličtinou viz např. Electric Word, May/June 1990, kde se počítá s korpusem v rozsahu 100 mil. slovních forem). Korpus bude postupně vznikat konverzí typografických textových souborů získávaných .z tiskáren a nakladatelství na ASCII soubory a vhodnou úpravou korpusů již existujících (např. ve VÚMSU).
2. úroveň **datové báze typu konkordance** obsahující hesla s kontexty a frekvenčními údaji a navazující na retrogradní slovník vytvořený v ÚJČ a rovněž i na soubor sémát vytvořený v ÚVTEI (Smetáček). Tato hesla mohou mít charakter „polotovarů“ a nemusí být zpočátku nijak homogenní.
3. úroveň **datové báze konkrétních slovníkových hesel** klasického typu (např. jako v SSJČ) nebo nového mezinárodního typu, která se předpokládají pro nové slovníky vytvářené v rámci evropského sdružení LANGUAGE INDUSTRIES. Tato báze by měla vzniknout buď sejmutím SSJČ pomocí scanneru⁴, nebo přepsáním SSJČ do počítače.
 - K tomu přistupuje pozitivní skutečnost, že *Slovník spisovné češtiny pro školu a veřejnost* (SSČ, ACADEMIA, Praha, 1987) existuje již v počítačově čitelné podobě a může tedy posloužit jako východisko pro další práci.
 - Pro práci s datovou bází slovníkových hesel není zatím momentálně k dispozici vhodný softwarový prostředek, tj. *specializovaný databázový systém*, který by byl okamžitě a bez výhrad použitelný pro budování české lexikální databáze.
4. Schůdné řešení pro všechny uvedené úrovně poskytuje výše citovaný projekt TEI (Text Encoding Initiative, C. M. Sperberg-McQueen & Lou Burnard, 1990), který umožňuje ukládat české texty, konkordance i slovníková hesla v textové podobě do běžných textových souborů a vnitřní hierarchie uvnitř hesel vyznačovat technikou SGML (STANDARD GENERALIZED MARKUP LANGUAGE).

lygrafia (Svoboda) a Spektrum (Tisk) a též se jedná o poskytnutí již existujících korpusů, např. ve Výzkumném ústavu matematických strojů.

⁴ Počítačovní odborníci vyslovují však o schůdnosti této cesty značné pochybnosti vzhledem k velké typografické složitosti slovníkových textů. Pro daný účel pokládají za neefektivnější slovník přepsat, ev. na smlouvu přepisát.

Uvedené datové báze poslouží jako zdroje pro vytvoření **lexikální databáze současné češtiny**, která pokryje celou slovní zásobu moderní češtiny a bude představovat strukturovanou zásobárnu údajů o jednotlivých vlastnostech českých lexikálních jednotek, a proto musí přirozeně obsahovat podstatně více informace než kterýkoli existující český slovník. Z dobře koncipované a vybudované lexikální databáze bude pak možno zadáním vhodných kombinací třídících kritérií vytvářet různé typy slovníků, s to i značně specializovaných.

Běžně mluvená čeština a české a moravské dialekty

Databáze běžně mluvené češtiny a českých a moravských dialektů by měly mít podobnou strukturu jako databáze současné češtiny a měla by také vznikat z podobných datovýchází, tj. z báze mluvené češtiny, z báze nářečních textů, nářeční konkordance a odpovídajících slovníkových hesel.

Starší čeština

Lexikální databáze starší češtiny by měla zahrnovat jak staročeskou slovní zásobu, jak je obsažena v existujících staročeských slovnících, tj. např. ve *Staročeském slovníku*, ACADEMIA, Praha, díl 1–4, 1903–1984 a *Malém staročeském slovníku*, SPN, Praha, 1978, tak i starší češtinu z období baroka a obrození, což jsou úseky dosud nezpracované.

I zde se jako nejschůdnější postup jeví budování tří výše uvedených typů datovýchází, tedy textové, konkordační a slovníkové.

Biblická datová báze

V současné kulturní a politické situaci je zřejmé, že daný projekt bude jako rvou přirozenou část zahrnovat i *českou biblickou lexikální databázi* vycházející z nového ekumenického překladu Bible do češtiny. Český biblický korpus existuje již v počítačově čitelné podobě získané konverzí typografických souborů. Bez začlenění biblické datové báze do počítačového fondu češtiny by naše kulturní dědictví v oblasti české slovní zásoby nebylo zachyceno úplným a vyčerpávajícím způsobem. Jde ostatně o starý dluh v naší lexikografické tradici. Překladatelská skupina církve československé evangelické již vyjádřila své pozitivní stanovisko.

Doplňující datové báze

Kromě uvedené triády obsahující *současnou češtinu, starší češtinu a české a moravské dialekty* pokládáme za potřebné začít systematicky vytvářet další tři nebo čtyři datové báze, které budou vhodným způsobem propojeny a mohou sloužit při vytváření lexikální databáze současné češtiny a také představovat zásobárny dat pro další výzkum.

Patří mezi ně:

1. **terminologická lingvistická databáze** obsahující údaje o lingvistických termínech získaných z jazykovědné literatury a periodik. Bylo by velmi žádoucí, aby

na vytváření této databáze se podílela všechna bohemistická, pracoviště v České republice.

2. **slovník českých kořenů a kmenů** a na něm založený derivační generátor a morfologický analyzátor. Je vcelku zřejmé, že tyto slovníky a programy mohou při vhodné implementaci fungovat jednak jako moduly uvnitř systémů pro zpracování přirozeného jazyka a jednak jako zdroje dat pro vlastní lexikální databázi. Díky své speciální povaze by však měly existovat i jako autonomní jednotky (moduly).
3. **úplný rejstřík (index)** všech slov a slovních forem obsažených ve všech právě uvedených databázích.
4. **úplný rejstřík (index)** všech slov a slovních forem obsažených ve všech právě uvedených databázích doplněný o veškeré **frekvenční údaje**. Tento rejstřík představuje východisko pro úplný **frekvenční slovník** češtiny.

Počítačový fond češtiny je koncipován jako plně *univerzální*, a má tedy umožňovat jak všestranný teoretický výzkum, tak i nejrůznější aplikace. Konkrétněji tedy půjde o možnosti konečně adekvátněji studovat úzus, o textovou analýzu, tvorbu nových gramatik a učebnic i konečně – v neposlední řadě – o prohloubené pokrytí paradigmatické a syntagmatické slova a jeho popis v nejrůznějších typech slovníků, především v *novém standardním slovníku českého jazyka*.

Hardware a software

Základní varianta

1. pro ukládání a zpracování textů a slovníků je potřeba počítat s jedním centrálním pracovištěm vybaveným hlavním počítačem typu RS6000 (IBM), SUN 4 nebo MIPS 3000 s operačním systémem UNIX, s vnitřní pamětí od 8 MB a diskovou pamětí od 1 GB výše (včetně standardního vybavení) (50 000 DM)
2. celé lexikografické pracoviště je potřeba vybavit lokální sítí ETHERNET tvořenou 10–12 počítači PC AT 386 SX (s pevnými disky 80 MB) (36 000 DM)
3. pro celé pracoviště je nezbytná výkonná laserová tiskárna se systémem PostScript (20 000 DM)
4. lexikografické pracoviště se neobejde bez standardního kancelářského vybavení včetně 1–2 výkonných kopírovacích přístrojů (jeden s formátem A3) plus dvě pevné telefonní linky a fax (12 000 DM)
5. pro archivování korpusů, konkordancí, slovníků a vytvořených lexikálních databází musí být lexikografické pracoviště vybaveno přepisovatelnými magnetooptickými pamětmi typu WORM a také jednotkami CD ROM (15 000 DM).

V oblasti software je potřeba počítat s následujícím vybavením:

1. soubor programů tvořících tzv. LEXIKOGRAFICKOU PRACOVNÍ STANICI. Jsou to programy, které umožňují:
 - sestavovat z existujícího korpusu konkordance
 - získávat synonymické řady a popřípadě i hierarchie
 - poskytovat lexikografovi veškeré údaje potřebné při vytváření slovníkového hesla, patří k nim programové systémy jako WORDCRUNCHER, OXFORD CONCORDANCE PROGRAM (OCP) a/nebo LEXICOGRAPHER WORKSTATION (Calzolari, Picchi, 1990) aj.,

2. velký systém pro hlavní počítač a program PAT (nahrazující databázi a vyhledávací programy pro texty a slovníky) plus další programy získané v rámci výše citovaného projektu TEI.
3. OPERAČNÍ SYSTÉMY:
 1. operační systém MS-DOS pro práci na jednotlivých počítačích (uvnitř lokální sítě),
 2. operační systém UNIX pro hlavní počítač sítě.
4. vybrané TEXTOVÉ EDITORY a TEXTOVÉ PROCESORY pro práci s českými textovými soubory na jednotlivých počítačích, např. WORDPERFECT nebo ζ SED, T602, UNIX EMACS aj.
5. DATABÁZOVÉ SYSTÉMY – v počáteční fázi nebude potřeba pracovat s žádnými speciálními systémy (např. DBASE, FOXBASE či později SEZAM pod systémem MS-DOS), pod systémem UNIX připadá v úvahu systém INFORMIX.
6. programy pro TRANSFORMACI typografických souborů na textové, z nichž bude tvořen korpus.
7. programy pro ANALÝZU a VYHLEDÁVÁNÍ (systémy typu PAT) a ZPRACOVÁNÍ slovníkových hesel v počítačově čitelných slovnících.
8. programy pro MORFOLOGICKOU ANALÝZU (lemmatizátory) a SYNTAKTICKOU ANALÝZU češtiny.
9. vhodný publikační (DTP – desk top publishing) systém. Za takový systém pokládáme \TeX , který plně vyhovuje předpokládaným potřebám projektu a patří do *public domain*).

Celkový odhad finančních nákladů na projekt činí 3,0–3,5 mil. Kčs.

Předpokládané výstupy a jejich využití

Počítačový fond češtiny bude poskytovat řadu výsledků vhodných pro využití v oblasti typografie, v redakcích, nakladatelstvích a všech institucích, v nichž jde o zpracování českých textů ve větším měřítku:

1. **Data**
 - reprezentativní korpus češtiny vytvořený použitím techniky SGML (viz výše)
 - různé typy konkordancí budované rovněž s použitím značek v rámci techniky SGML
 - datové báze a posléze i slovníky v počítačově čitelné podobě
2. **Procedury a programy**
 - Na základě uvedených dat lze získávat specializované a pokud možno úplné podklady pro budování:
 - dokonalejších automatických korektorů schopných korigovat nejen překlepy, ale i chyby gramatické (např. shodu a chybné vazby) a stylistické
 - co nejuplnějších programů pro české dělení slov
 - programové moduly umožňující komunikaci s počítačem v přirozeném jazyce (komunikace s datovými bázemi), automatické gramatiky pro strojový překlad apod.
3. **Slovníky a gramatiky**
 - Vytvořená data umožní také připravovat různé typy slovníků v textové i strojově čitelné podobě, orientované na různé skupiny uživatelů

- nové gramatiky založené na kvalitních datech a dostatečně úplném materiálu
- dokonalejší učebnice mateřského jazyka pro jednotlivé typy škol zachycující co nejpřesněji současnou normu naší mateřštiny
- kvalitní příručky a učebnice pro výuku češtiny jako cizího jazyka.

Odbornými garanty projektu **Počítačového fondu češtiny** jsou pracovníci vysokých škol a Ústavu pro jazyk český ČSAV, konkrétně doc. dr. František Cermák, DrSc., FF UK a ÚJČ ČSAV Praha, prof. dr. Petr Sgall, DrSc., FF UK Praha, dr. Karel Pala, CSc., FF MU Brno, doc. dr. Milada Hirschová, CSc., FF UP Olomouc, RNDr. Jan Králík, ÚJČ ČSAV Praha, dr. Eva Hajičová, DrSc., MFF UK Praha, doc. dr. Karel Kučera, CSc., FF UK Praha, RNDr. Jan Hajič, MFF UK Praha, dr. Věra Schmiedtová, ÚJČ ČSAV Praha.

V současné situaci se však dost dobře nelze spoléhat na to, že by ČSAV poskytla finanční prostředky pro projekt v dostatečné míře, proto se obrácíme na domácí i zahraniční sponzory s žádostí o finanční podporu **Počítačového fondu češtiny**. Věříme, že i členové ČTUG projeví plné pochopení pro naše úsilí. (*Příspěvky lze posílat na bankovní účet České státní spořitelny 0800-2044345-018, konstantní symbol 0558.*)

(Dr. Karel Pala, CSc.)

Music \TeX —Sazba notových partů pomocí \TeX u

Úvod aneb jak jsem k sazbě not přinucen byl

Letos na začátku června jsem se zúčastnil celostátní konference z teorie grafů. Setkal jsem se zde s problémy, které trápí především nečtenáře \TeX bulletinu – a totiž jak a čím psát matematické články. Jako všichni, kteří propadli \TeX u, jsem reagoval i já – hodinovou přednáškou na téma \TeX . A abych všechny přesvědčil o nedostižných kvalitách \TeX u, tvrdil jsem, že \TeX umí „prostě všechno“. Na dotaz, zda umí sázet i noty jsem odpověděl svým vlastním citátem: „ \TeX umí vše, jenom se musí vědět jak“. Na toto zareagovali mí posluchači a tak jsem dostal za úkol vysázet \TeX em „Hymnu teorie grafů“ (jedna z ukázek).

Nastala doba shánění informací, kdo, kdy a s jakým úspěchem se věnoval sázení not. Objevil jsem jediný balík: maker, nazvaný svým autorem D. Taupinem Music \TeX . S ním se mi po menší námaze podařilo vysázet už zmíněnou skladbu a tak jsem souhlasil s návrhem podělit se s vámi o svoje zkušenosti. Jsou to zkušenosti člověka naprosto neznalého notového zápisu, bez jakéhokoli hudebního sluchu. Toto mějte, prosím Vás, na paměti při posuzování mého článku.

K čemu lze použít Music \TeX

Music \TeX je „*public-domain*“ balík maker, určených k sazbě polyfonní a instrumentální hudby. Pomocí něj jste schopni vysázet šest nástrojů (přičemž hlas se počítá za dva nástroje - první pro hudbu, druhý pro text). Kódování pro \TeX je

celkem logické, pochopitelné i pro hudebně vzdělaného, leč Tem netknutého člověka. (Předložil jsem se Taupinův návod jednomu ze svých přátel, a ten tvrdí, že by si na sazbu troufal.)

Popis MusicT_EXu

V této části popíšu obsah distribuované verze a způsob práce.

Obsah veřejně dostupné verze

Nejdůležitějšími soubory v MusicT_EXu jsou následující soubory `musictex.tex`, `musicnft.tex` a `musicpln.tex`. Posledně dva jmenované obsahují definice nota dalších notových značek. `musicnft` používá speciální fonty `musicn20`, `bean16` a `slur16`; `musicpln` vytváří noty pomocí standardních fontů `linew10` a `circlew10`. Rozdíl ve kvalitě jak vzhledu, tak čitelnosti je patrný.

Tyto tři soubory doplňují další: `musicadd.tex` pro sazbu více než šesti nástrojů, `genpctex.tex` modifikující makra $\mathcal{A}\mathcal{M}\mathcal{S}$ -T_EXu a soubory modifikující výstup podle typu tiskárny. V balíku nechybí ani popis fontů `music`, `beam` a `slur` pro METAFONT. Návod v angličtině má šestnáct stránek a zřejmě obsahuje většinu informací, mně připadal trochu nedostačující. Postrádal jsem přehledný seznam symbolů i více kratších příkladů objasňující jednotlivé pojmy. Mimo anglický návod existuje i francouzský návod, který ale není přesnou kopií anglického.

Zbytek souborů obsahuje příklady – přepisy známých skladeb. Z nich se člověk může mnohému naučit, ale za ‘lehce čitelné’ bych zdrojové texty pro MusicT_EX rozhodně neoznačil.

Způsob práce s MusicT_EXem

MusicT_EX je nadstavba PlainT_EXu, počtem vyžadované paměti se prý blíží L^AT_EXu. Je ale postaven dosti čistě, takže jej lze použít i nad $\mathcal{A}\mathcal{M}\mathcal{S}$ -T_EXem. Pravda, pak už se do operační paměti mého počítače celá strana not velikosti A4 nevešla. MusicT_EX sází najednou svislý sloupec not, vždy pro všechny nástroje současně. Způsob kódování je blízký hudebnímu způsobu čtení not, noty se zapisují od spodu nahoru. Pro práci s MusicT_EXem je tedy nejpodstatnější makro

```
\notes ... & ... & ... \enotes,
```

kde znak A je používán k oddělení not jednotlivých nástrojů. Tento znak je rezervován pro sazbu not a chráněn před jiným použitím pomocí `\catcode`.

Pokud je jeden nástroj rozepsán na více notových osnov [anglicky ‘staff’], jsou jednotlivé osnovy oddělovány znakem |. Tedy pokud budu psát hudbu pro piano a housle, mohu použít makra

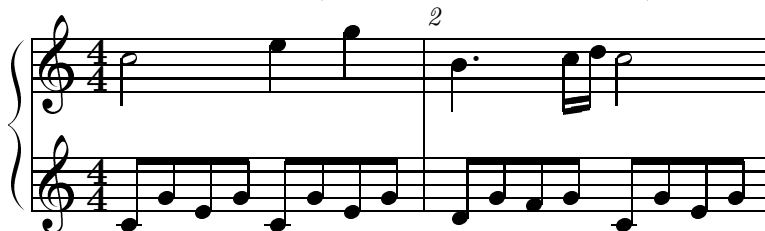
```
\notes ... | ... & ... \enotes
```

pro každý sloupec notového záznamu.

Jednotlivé noty se kódují dosti logicky, pro nehuďebníka snad ještě lépe než pro hudebně vzdělaného člověka. Výšku tónu (svislá pozice noty) je dána písmeny a ... z a A ... N. Délka tónu je kódována písmeny odvozenými z anglických názvů

(*wh* pro celou, *q* pro čtvrtovou notu ...). K dispozici jsou noty až 64-tinové. Otočení „nožičky“ nahoru nebo dolů určují písmena *u*[*up*] a *l* [*low*]. Tedy poloviční dolní *C* otočeno nahoru zapíšou *\hu c*, osminové *C* o oktávu výše zapíšou *\cl i*.

Myslím si, že je na čase ukázat už konkrétní příklad (pan Taupin mi snad promine, že použiji jím připravený). Jedná se o Mozartovu sonátu (Mozart, K545):



Zdrojový text vypadá takto:

```



```

MusicTeX nekontroluje sám, zda to, co sází je hudebně správně (mám na mysli délku taktu, správné mezerování, správné otočení nožičky nahoru nebo dolů). Vykonnává tupě jen sazečovy příkazy, zato s TeXovskou dokonalostí.

Podle ohlasů z mezinárodních elektronických sítí se v MusicTeXu dají ještě najít drobné chyby, ale jinak se, myslím, jedná o velmi užitečný balík maker.

Co zatím asi určitě chybí je nějaký grafický WYSIWYG preprocesor, ve kterém by se daly noty lehčeji editovat, který by kontroloval správnost zápisu a který by případně i sám pravá, děl některá jednodušší rozhodnutí týkající se správného mezerování. *Námět pro diplomovou práci budoacího TeXperta ???*

Tuto stručnou informaci o Music \TeX u zakončím ukázkou méj práce na „Hymně Teorie grafů“.

Hymna Teorie grafů

Zdeněk Ryjáček / Bohdan Zelinka

Transkripce pro \TeX P. Sekanina (1991)

The image shows a musical score for a hymn. It consists of four staves of music in G-clef and 4/4 time. The lyrics are written below the notes. The score is numbered 1 through 8. The lyrics are: Přes Pre-go-lu se-dm mos-tů stá-lo, na svou do-bu ne-by-lo to má-lo, krá-lo-več-tí rad-ní hr-di by-li, že si ty-to mos-ty pos-ta-vi-li.

Příjemné chvílky při krocení not přeje všem odvážným čechoslovákům autor této poznámky.

(Podle článku Daniela TAUPINA napsal a převyprávěl *Pavel SEKANINA*)

Konkursní řízení

ζ TUG pořádá výběrové řízení pro zakoupení editoru upraveného pro přípravu textů systémem \TeX v češtině a slovenštině.

Editor by měl umožňovat: snadnou upravitelnost pro oba jazyky, měl by být lehce ovladatelný, maximálně přizpůsobitelný uživateli (konfigurovatelnost pro většinu běžně užívaných grafických karet, konfigurovatelnost klávesnice, snadná tvorba dostatečného počtu maker, dostupná cena při skupinové licenci pro členy sdružení, široká škála operací, rychlost).

K nabídce je třeba poskytnout záruku na odstraňování nedostatků a případné úpravy (za dodatečnou úplatu) po dobu cca dvou let. Zájemci nechť kontaktují

Annu Najmanovou, MU UK, Sokolovská 83, 186 00 PRAHA 8 –
Karlín, tel. 231 60 00, linka

Termín k předložení: se třemi exempláři manuálu do 31.ledna 1992 na uvedenou adresu (ζ TUG).

V prvním čísle $\text{T}_{\text{E}}\text{X}$ bulletinu byl uveřejněn seznam **dobrých vlastností** editoru vhodného pro přípravu textů pro $\text{T}_{\text{E}}\text{X}$. Nyní ho s drobnými úpravami přetiskujeme, je však třeba chápat, že jde o ideální editor. Poznamenejme, že následující výčet nemusí uspokojovat každého, ale od vydání prvního čísla $\text{T}_{\text{E}}\text{X}$ bulletinu až do dnešních dnů jsme neobdrželi žádné důležitější doplňky. Nezabýváme se např. „lepivostí“ kláves (nastavení časových poměrů pro opakování znaků při déle stlačené klávese a pod.), jde o pohled obvyčejného **T_EXového** uživatele, za které se považujeme.

Kromě „obvyklých“ povelů je vhodné, aby editor měl následující funkce (mezi obvyklé povely řadím pohyby kurzorem, mazání znaků, slov, . . . , práce s bloky, vyhledávání a náhrady řetězců znaků, atp.):

- Skok na řádek se zadaným číslem.
- Skok na začátek, resp. konec vyznačeného bloku.
- Kopie vyznačeného bloku (pokud není blok vyznačen, tak celého řádku, ve kterém je kurzor) do dočasného zvláštního bufferu (spec. pro tento účel).
- Uložení vyznačeného bloku (pokud není blok vyznačen, tak celého řádku, ve kterém je kurzor) do dočasného zvláštního bufferu a jeho vymazání ve zdrojovém souboru.
- Kopie obsahu dočasného zvláštního bufferu do místa (nebo za řádek, pokud byl do dočasného bufferu uložen celý řádek a nikoliv blok) ve kterém je kurzor.
- Možnost kopírovat znak po znaku obsah předchozího řádku do řádku, ve kterém je kurzor.
- Automatické přerazování slov při psaní přes určitý sloupec a jeho vypínání (wordwrap).
- Možnost uvedení řádku do tvaru, v jakém byl v okamžiku vstupu kurzoru do tohoto řádku (po provedení jakýchkoli operací v tomto řádku).
- Obnova vymazaných řádků (jejich postupné vkládání před řádek, ve kterém je kurzor — alespoň 20 naposledy vymazaných řádků).
- Hledání párových závorek stejného typu ke kulatým, hranatým a složeným závočkám a hledání párových apostrofů.
- Možnost načíst předdefinovaná makra (odhaduji asi 150 maker v průměrné délce 18 znaků, u některých maker i větší délky) v okamžiku startování editoru (jméno souboru maker by mělo být zadavatelné jako parametr příkazové řádky a uživatel by měl mít možnost relativně snadno tato makra vytvářet a měnit).
- Možnost definovat makra během práce s editorem, ukládat je do souboru na disk a opět je číst.
- Ukládat bloky do dočasných (slovem „dočasný“ je míněna doba než opustíme editor) „pojmenovaných“ bufferů (tj. při ukládání jsem dotázán na jméno bufferu — něco jako zápis bloku na disk do souboru, ale bez použití disku) a vkládat text z pojmenovaných bufferů do místa s kurzorem (ev. s přesunem kurzoru na konec vloženého textu).
- Mít možnost otevřít alespoň 3 soubory (tj. zdrojový soubor, chybový soubor a alespoň jeden další).
- Automatické natahování alespoň dvou (ale raději tří) souborů při startování editoru.
- Umožňovat dělit obrazovku na více oken (alespoň na 2).

- Při rozdělení obrazovky na více oken možnost rolovat oběma směry v předcházejícím okně (předcházejícím oproti oknu, ve kterém je kurzor, tj. které je aktivní), aniž by kurzor přecházel do jiného okna.
- Při rozdělení obrazovky na více oken možnost dělat ZOOM aktivního okna (tj. dočasné roztažení na celou obrazovku a uvedení do původního stavu).
- Při nataženém chybovém a zdrojovém souboru a rozdělení obrazovce na dvě okna (v aktivním okně zdrojový a ve druhém okně chybový soubor) nalézt v chybovém souboru další chybu, zobrazit ji a ve vstupním souboru umístit kurzor do řádku, ve kterém je chyba (jedná se o nalezení vykřičníku na začátku řádku v chybovém souboru).
- Umožňovat automatické uschovávání editovaného souboru po určitém čase nebo po vstupu určitého počtu znaků.
- Uživatel by měl mít možnost konfigurovat významové kombinace kláves podle svých zvyklostí, aby nebyl nucen učit se nové ovládání.
- Možnost určit pořadí a názvy extenzí souborů pro natahování. V případě, že je uveden pouze název souboru, bude natažen ten soubor, jehož extenze je uvedena v seznamu „nejvýše“. Pokud neexistuje žádný soubor s extenzí uvedenou v seznamu, založí se nový soubor zadaného jména (bez extenze).
- Při všem, co je uvedeno výše, by editor měl být malý a rychlý (rychle se spouštět a rychle provádět často užívané časově náročné operace — např. natahování souborů, hledání a nahrazování řetězců znaků, přesun bloků, a pod.).

(Oldřich Ulrych)
e-mail: ummou@csearn

(Jiří Veselý)
e-mail: ummjv@csearn

Vytváření rejstříků: Část 1

Lincoln Durst

Tři předchozí příspěvky této řady jsou v dřívějších číslech *Tugboatu* [10#3 (November 1989, strany 390–394), 11#1 (April 1990, strany 62–68) a 11#4 (November 1990, strany 580–588)]. Tato část je víceméně na předchozích nezávislá; přístup k problému je nicméně v podobném duchu. Ve skutečnosti některé myšlenky inspirovaly vývoj popsany v předchozích částech.

Při vytváření rejstříků pomocí $\text{T}_{\text{E}}\text{X}$ u jsou nutné přibližně čtyři hlavní kroky stejně jako za starých časů, kdy se rejstříky vytvářely ručně; a to 1) výběr položek, které jsou rozumnými kandidáty na rejstříkové položky, 2) výběr vhodného druhu jejich uspořádání (abecedního pro pořádání slov, číselného pro stránkování), 3) spojení, anebo vyloučení duplikátů, odstranění nesrovnalostí mezi položkami z různých částí knihy, které se po setřídění třeba octnou blízko sebe, a 4) uspořádání rejstříku do podoby, ve které je ho možno vytisknout. Doba se ve skutečnosti příliš nezměnila;

tyto kroky uvedené v přirozeném pořadí jsou stále i přes technický pokrok uvedeny v pořadí rostoucí obtížnosti.

Označit vhodné termíny pro rejstřík a zapsat je do souboru v pořadí, v jakém se objevují v textu, je snadné. Uspořádání tohoto souboru však není tak jednoduché jako uspořádání souboru maker pro položky v bibliografii, protože jsou spojeny s čísly stránek, a pokud nejste dost opatrní a mazání, jednoduché seřídění podle ASCII kódu vám třeba může dát

$$1 < 11 < 111 < 2 < 21 \dots$$

(znalci to nazývají lexikografickým uspořádáním). Po automatickém seřídění nastanou další problémy, které vyplývají z rozdílů psaní velkých a malých začátečních písmen, jednotného a množného čísla apod. A nakonec je potřeba rozlišit hlavní a vedlejší položky.

Záleží na vkusu, zda poslední problém budete řešit jako první, nebo jako poslední. Problém je to filosofický, protože se redukuje na to, zda má být logické rozlišení určeno před začátkem psaní (tzn.: že problém co uspořádat či přerovnat je tu zcela abstraktní otázkou není co pořádat]), nebo zda se tím budeme trápit až po sestavení seznamu položek. Dlouhodobí čtenáři *TUGboat* si mohli všimnout, že zdejší varianty „slepice a vejce“ jsou jen opakováním dilematu předloženého Richardem Southallem před šesti lety [5#2, November 1984, strana 80], zda je vhodné (nebo možné) vytvořit zdrojový text ještě předtím, než je jasná jeho grafická úprava při vtištění.

Tato část je věnována prvním dvěma krokům a pojednává jednak o vytvoření souboru poznámek, jednak o některých problémech, se kterými je možno se setkat při jejich výběru.

Vytvoření souboru poznámek. Zde si vezmeme zjednodušenou verzi Knuthova rejstříkového makra z *TEXbook* popsanou v Appendixu E.⁵

Knuth stanovil čtyři druhy položek svého rejstříku, např.

Arabic, by, \char, <dimen>.

Obyčejný smrtelník obvykle vystačí jen s jedním druhem položek. Jisté bude jednodušší to pochopit, jestliže omezíme pozornost jen na první z Knuthových čtyř případů. Jakmile porozumíte základnímu principu, pusťte se klidně vlastní cestou, pokud se náhodou rozhodnete napsat knihu o T_EXu.

Knuth to dělá tak, že označuje v textu souboru termíny, které chce zahrnout do rejstříku (*The T_EXbook*, strany 415–416, 423–424). Označování se může provádět během psaní, anebo až při kontrole textu. Knuth o nich mluví jako o „index reminders“, protože tyto značky nebudou ještě automaticky tvořit konečné položky rejstříku; mnohé budou teprve editovány v dalším průběhu konstrukce rejstříku. Předpokládejme, že „Gauss“ je zmíněn v textu, potom slovo „Gauss“ bude v textu vyznačeno, i když odpovídající položka rejstříku se nakonec změní na

Gauss, Karl Friedrich (1777–1855) . . .

Knuthův způsob označení termínů v textu je buď `~{Gauss}`, nebo `^^{Gauss}`; v prvním případě je slovo „Gauss“ vtištěno v textu, zatímco ve druhém ne, ačkoliv

⁵ Jiné způsoby přípravy rejstříku byly popsány v dřívějších *TUGboatech*, počínaje prvním svazkem u září 1980 (1#1, Winograd & Paxton, Appendix A, str. 1–12, případně později v listopadu 1989 (10#3, David Salomon, str. 394–400). Pro čtenáře, které zajímají možné alternativy, by mohlo být srovnání těchto dvou pramenů užitečné.

je bráno v úvahu při vytváření rejstříku. Např. zdrojový text pro obr.1 obsahuje následující pasáž:

```
By means of  $\text{\^{\{stops\}}$  the performer has
within his pover a number of combinations
for varying the  $\text{\^{\{tone\}}$  and
 $\text{\^{\{power, dynamic\}}$ 
dynamic power.
```

V *TEXbooku* se položky druhé kategorie, které mají dvě „stříšky“, $\text{\^{\dots}}$, označují jako „tiché“. (To je důvod — viz níže — pro umístění tohoto tichého vstupu právě před slovo (slova), jehož se to týká; jejich oddělení na zvláštní řádek může být pro čtenáře textu na obrazovce poněkud snazší, protože všechny ostatní řádky pak může přeskočít).

Knuth určil kód, kterým se poznámky zapisují do souboru (v pořadí, v jakém se slova vyskytují v textu), a kód, který provádí jejich výpis na okraji stránky, jak ukazuje výsledek na obr. 1. Zde je zkrácená verze [*The TEXbook*, strany 415, 423]:

```
%%% index.rem, first fragment %%%
\newinsert\margin \dimen\margin=\maxdimen
\count\margin=0 \skip\margin=0pt
\nevf\ifsilent \newwrite\inx
\immediate\openout\inx=\jobname.inx
\def\specialhat{%
  \ifmode\def\next{\^}%
  \else\let\next=\beginxref
  \fi
  \next}
\catcode'\^=\active
\let ^=\specialhat
```

Nejprve určíme vsuvku (\insert), která bude obsahovat seznam na okraji. Rozměr této vsuvky může být výška celé stránky (\maxdimen), ale ve vertikálním prostoru zabírá nulovou výšku (\count\margin=0), tedy nezabírá žádnou textovou plochu běžné stránky: jak jsme už nejednou viděli, \TeX může bez viditelného odporu přijmout některá tvrzení, která si mohou hrubě odporovat. Pokud se chcete o vsuvkách (insercích), jež jsou v *TEXbooku* pojednány s charakteristickou stručností, dozvědět něco více, nahlédněte do článku Davida Salomona v listopadovém vydání *TUGboatu* z roku 1990 (D. Salomon, 11#4, stránky 588–605). Následuje \if-příkaz , který je určen k rozlišení mezi jedno- a dvoustříškovými položkami, a poté se otevře soubor, který obsahuje jednotlivé položky. Nakonec je \^ definován jako aktivní symbol a \specialhat jako jeho náhrada. Je samozřejmě nezbytné zdůraznit skutečnost, že \^ může být v matematickém módu použit obvyklým způsobem, takže v případě splnění podmínky \ifmode je \specialhat obvyklá \^ , pokud ovšem nejsme v matematickém módu, stane se z něj \beginxref , následující položka v naší agendě [*dtto*, str. 423].

```
%%% index.rem, second piece %%%
\def\beginxref{\futurelet\next
\beginxrefswitch}
\def\beginxrefswitch{%
  \ifx\next\specialhat
```

```

\let\next=\silentref
\else
\silentfalse\let\next=\xref
\fi
\next}
\def\silentref^{\silenttrue\xref}

```

Tady musíme být opatrní a lstiví. \TeX našel $\hat{}$ mimo matematický modus, takže $\backslash next$ je nyní $\backslash beginxref$. První otázka, kterou si musí položit, je, zda následuje další $\hat{}$. Uvažme uvedené dva případy, $\hat{}\{Gauss\}$ a $\hat{}\{Gauss\}$: v prvním případě dalším znakem, jenž \TeX najde, bude $\{$, zatímco v druhém případě to je „ $\hat{}$ “, tj. $\backslash specialhat$. Příkaz $\backslash futurelet$ (viz *The \TeX book*, strana 307) \TeX u umožňuje zaň dosadit $\backslash next$; pak dojde k expanzi $\backslash beginxrefsvitch$. Jestliže nastal druhý případ, příkaz $\backslash silentref$ odstraní $\hat{}$, nastaví přepínač $\backslash ifsilent$ na pravdivou hodnotu a zavolá příkaz $\backslash xref$; v druhém případě se rovněž volá $\backslash xref$, ale podmínka $\backslash ifsilent$ zůstává nepravdivá. Diskuse o $\backslash futurelet$ na str. 207 je jistě příliš stručná; čtenáři, kteří by rádi viděli i jiné příklady jeho použití, mohou konzultovat poučný článek Stephana v. Bechtolsheima v *TUGboat* **9#3** December 1988, na str. 276 a dalších.] V každém případě se nejprve expanduje příkaz $\backslash beginxrefswitch$, po němž je správně nastaven přepínač $\backslash ifsilent$ a zavolá se $\backslash xref$.

```

Nyní je již čas na  $\backslash xref$ :          %% index.rem, third part %%
\def\xref#1{\def\text{#1}%
\let\next=\text\makexref}
\def\makexref{%
\ifproofmode\insert\margin{%
\hbox{\marginfont\text}}%
\xdef\writeit{\write\inx{%
\text\space!0\space
\noexpand\number\pageno.}}%
\writeit
\else
\ifhmode\kern0pt\fi
\fi
\ifsilent\ignorespaces\else\next\fi}

```

První dva řádky zde nahrazují patnáct (mnohem delších) řádek kódu uprostřed str. 424, kde dochází ke čtyřnásobnému větvení podle čtyř typů rejstříkových položek, s kterými Knuth kouzlí. Všimněte si, že dvě kopie textu v závorkách jsou vytvořeny pomocí $\backslash xref$, jedna z nich ($\backslash text$) je určena pro rejstříkový soubor a poznámku na okraj textu a druhá ($\backslash next$) je tou položkou, která se objeví v textu v případě $\backslash silentfalse$. Právě popsané kroky se uskutečňují pomocí $\backslash makexref$. Všimněte si $\backslash space!0\space$ v definici; to se používá k oddělení jednotlivých terminů pro rejstřík od čísla stránky, na které se položka objevila. Jediným kritériem pro výběr vhodného znaku pro oddělovač je, že by to mělo být něco, co se nikdy neobjeví jako kandidát pro zápis do rejstříku. Tady vyhovuje libovolná dostatečně nepravděpodobná kombinace znaků. Knuth pro své čtyři případy používá $!0$, $!1$, $!2$ a $!3$ a jeho označení se s naším shoduje. Tečka za $\backslash pageno$ označuje konec položky. Jestliže vaše číslování stránek obsahuje tečky (viz níže), měli byste ukončit svoji položku některým jiným znakem (např. $:$). Dvě jemnosti: (1) $\backslash kern0pt$ potlačuje dělení slov, pokud nejsme

v korekturním módu (`\ifproofmode`), to pro shodu s výsledkem v případě vsuvky (viz Appendix H, *The T_EXbook*, strany 454, 455). (2) `\ignorespaces` potlačuje každou mezeru, explicitní, nebo implicitní na konci položky „`^^Gauss`“, což znamená, že v textu

```
$1+i$ is a ^^Gauss gaussian prime.
```

nebo

```
$1+i$ is a
^^Causs
gaussian prime.
```

nemůže dojít k zalomení stránek mezi `^Caus` a `gaussian`: Tyto dvě položky jsou spolu pevně spojeny (jako `superlepidlem`, které ovšem nemá žádnou vůli).

K dokončení konstrukce obr. 1 musíme poněkud zacvičit s výstupem stránky, abychom dostali vsuvku na okraj, když jsme v korekturním módu. Knuthova verze je na str. 416, naše je o něco jednodušší, protože nezkoušíme dělat vše, co musel dělat on:

```
%%% index.rem, output routine %%%
\newdimen\pageheight \pageheight=\vsize
\def\onepageout{\shipout
  \vbox{\offinterlineskip
    \makeheadline\pagebody}\advancepageno}
\def\makeheadline{\vbox to 2pc{%
  \ifodd\pageno\rightheadline
  \else\leftheadline\fi\vfill}}
\def \pagebody{\vbox to\pageheight{%
  \ifproofmode\InsertNotes\fi
  \unvbox255}}
\output={\onepageout}
\newdimen\remkern \remkern=-8pc
\def\InaertNotes{\ifvoid\margin\else
  \rlap{\kern\remkern\vbox to 0pt
    {\box\margin\vss}}\fi}
```

Tím přepíšeme `\output` definovaný v `plain.texu`, neboť ho nahrazujeme příkazem `\onepageout`; příkaz `\offinterlineskip` skládá oba boxy (`\makeheadline` a `\pagebody`) těsně k sobě, aniž by ponechal mezi nimi jakýkoli prostor, a `\vbox255` je box, který obsahuje hotovou stranu textu. Příkaz `\InsertBotes` přidává obsah vsuvky `\margin`, je-li jaký. Konečně potřebujeme (víceměň) vhodný řez písma pro marginálie:

```
%%% index.rem, reminders in margin %%%
\font\remfont=cmtt8 \font\strutfont=cmr9
\newbox\rembox
\setbox\rembox=\hbox{\strutfont })
\def\remstrut{\vrule height 1\ht\rembox
  depth 1\dp\rembox width 0pt}
\def\marginfont{\remstrut\remfont}
```

Knuth dává tyto poznámky na pravý okraj, v našem případě jsme zvolili levou, protože pravá strana je již přepracována. Nepsaná moudrost profesionálních technických redaktorů (upravovatelů knih a jiných tištěných materiálů) opírající se o studii

čitelnosti říká, že tištěný text má obsahovat průměrně deset nebo méně slov v řádku. To dává pro šířku textu (`\hspace`) v případě desetibodových typů asi 24 cicer a nejvýše 30 cicer pro dvanáctibodové typy. To znamená, že jeden sloupec desetibodového textu na stránce formátu A4 by měl vždy mít levý a pravý okraj, jejichž celková šířka je skoro 10 cm, tj. téměř polovina šířky papíru, což nechává spoustu volného místa pro poznámky na obou stranách textu. Viz např. Southall, *dto*, str. 86. Rovněž *Designing with type* od Jamese Craiga (New York, Watson-Guptill; London, Pitman; opravené vydání 1980), str. 128. *Typography: How to make it most legible* od Rolfa F. Rehe (Carmel, Indiana, Design Research International; 5. vydání, 1984) obsahuje rozsáhlou bibliografii výzkumných zpráv týkajících se čitelnosti.⁶

Jako obvykle máme tu něco pro `prepare.tex`:

```
\newif\ifIndRemMark \IndRemMarkfalse
\def\IndexRemindersMarked{\IndRemMarktrue}
\newif\ifproofmode \proofmodefalse
\def\NriteIndexReminders{\proofmodetrue}
```

a něco pro `compose.tex`: `\ifIndRemMark\input index.rem\fi`

A navíc další dvě možnosti pro řídicí soubor:

```
\IndexRemindersMarked
\irriteIndexReminders
```

První volba vkládá do zdrojového souboru `index.rem`, aby se se mohl vypořádat se všemi `\specialhat` a druhá zapisuje označené položky do souboru a umožňují jejich vytištění na okraji textu v době, kdy je rejstřík připravován, ještě před závěrečným `TEX`ováním. (O souborech `\prepare.tex`, `\compose.tex` a řídicích souborech pojednává druhá část tohoto seriálu, viz výše.)

Varování. Kvůli jednoduchosti se v rejstříkových položkách kromě běžné interpunkce vyvarujte všech znaků, jež nejsou písmeny, nebo se dočkáte příšerných zážitků při třídění souboru. Nepoužívejte žádná diakritická znaménka, vlnky (~), ani řídicí slova libovolného druhu a za každou cenu se vyvarujte matematického módu. Tato opomenutí mohou být opravena, když se rejstříkový soubor upravuje do konečné podoby. V nezbytném případě je nutno psát

```
... described in detail in his
^^{Heiligenstaat Testament}
Hei\~li\~gen\~staat Test\~a\~ment. ...
```

(Viz též Dodatek dále.)

Uspořádání souboru položek. První věc, kterou musíme se souborem udělat, je jeho abecední uspořádání. Pokud jste nadáni stručností a čísla vašich stránek nemají víc než jednu číslici (nebo když se z nějakého důvodu stane, že všechny mají shodný počet číslic), je problém jednoduchý; použijte třídící utilitu, kterou poskytuje váš operační systém. Jinak bude patrně potřeba těžší kanón.

Kernighan & Ritchie ve své knize *The C programming language* (2. vydání), Prentice-Hall 1988 (odstavec 5.6, strany 108–110) ukazují, jak třídít soubory pomocí Hoarova rychlého třídění. Jestliže dáváte přednost Shellovu třídění, podívejte se do

⁶ Je možné, že v USA jsou vysoké školy, jejichž záměry vyžadují, aby diplomka měla obtížně čitelnou či dokonce vůbec nečitelnou formu? Dozajista boj proti volnému šíření myšlenek!

knihy Harbison & Steele, *C: A reference manual* (2. vydání), Prentice-Hall 1987, strany 210, 211, na smíšenou Knuthovu-Sedgewickovu-Harbisonovu-Steelovu verzi a vložte ji do K & R kódu. V každém případě není těžké upravit kód tak, aby části řádků předcházejících „!0“ byly uspořádány abecedně a části následující „!0“ numericky. Trik spočívá ve změně kritéria pro prohazování řádků, aby dva řádky byly zaměněny; jestliže řetězce písmen (a) jsou ve špatném pořadí, nebo (b) jsou stejné, ale čísla stránek nejsou správně seřazena. Když jste umínění a trváte na stránkování ve formě 1-12 [IBM, tady 1-2 < 1-12], 2.17 [Zenith, tady 2.2<2.17], K-9 [PCT_EX], atd., následky si poneseáte sami. Můžete je ovšem chápat jako výzvu.⁷

Jak už jsem naznačil, existuje jiný způsob, jak chytit kočku za ocas, pokud čísla vašich stránek jsou stará dobrá přirozená čísla. Naštěstí jsou ASCII kódy (na rozdíl od telefonních číselníků a klávesnic psacího stroje) uspořádány tak, že 0 předchází všechny ostatní číslice. Připsáním dostatečného počtu nul před `\pageno` tak, aby všechna čísla stránek měla shodný počet číslic, můžeme dosáhnout shody lexikografického uspořádání s číselným. Jestliže nemáte víc než tisíc stránek (máte-li, je třeba vám požeňnat), můžete použít následující kód

```
\def\Pageno{%
  \ifnum\pageno>99
    \noexpand\number\pageno
  \else\ifnum\pageno>9
    0\noexpand\number\pageno
  \else
    00\noexpand\number\pageno
  \fi\fi}
```

a dejte „`\Pageno`.“ do definice `\makehref` místo „`\noexpand\number\pageno`.“ S touto změnou můžete použít třídící utilitu, kterou nabízí váš operační systém, a po netřídění můžete odstranit teď už nadbytečné nuly.

Dodatek. Při jiné příležitosti navrhuje uvažovat kroky, s jejichž pomocí lze setříděný soubor vybraných položek přeměnit na docela dobrý rejstřík. Je třeba počítat s tím, že nějaká „ruční práce“ bude nutná, i kdyby šlo pouze o vložení druhého jména, dat či jiných podrobností, jež autor považuje za relevantní nebo důležité. Ale pořád zbývá dost rutinní práce, která může být zautomatizována. Navíc při troše chytrosti je vždy možno omezení týkající se zakázaných diakritických znamének a jiných řídicích příkazů uvnitř vybraných položek nějak překonat.

Poznámka. Současná verze zdroje, o němž je zmínka v předchozí části tohoto seriálu, spolu s již zmíněnými soubory obsahuje i soubor `index.rem` a zdrojový text pro obr. 1.

Je příjemné, jako ostatně vždy, poděkovat za pomoc, vedení a podporu všem, kteří nějak přispěli ke vzniku právě uvedené série článků. Ke všem obvyklým pochybnostem (o nichž jsem se tu již několikrát zmiňoval) musím vyjádřit svoji vděčnost

⁷ Menší výzvu představují tiskové chyby v pasážích citovaných na začátku tohoto odstavce. V K & R kódu nahraďte `maxlines` výrazem `MAXLINES`; ve výkladu H & S nahraďte „nejmenší číslo“ spojením „největší číslo“. (Chyby se objevily přinejmenším v prvních výtiscích obou těchto druhých vydání.)

Costovi Mylonasovi z Brown University a z Atén (Řecko), který mne vyzval, abych vytvořil soubor pokud možno jednoduchých a pružných prostředků, jež by mohly být užitečné pro ty autory, kteří mají chuť sedět u svých počítačů a psát knihy. Série mých učebních textů je výsledkem této výzvy. Doufám, že budou pro někoho užitečné, nebo alespoň poučné.

Lincoln Durst
46 Walnut Road
Barrington, RI 02806
lkd@math.ams.com

(Překlad: *Blanka Koubová*)

Kolik časopisů je vydáváno T_EXem v Československu?

Vydáváte časopis sázený T_EXem a kladete si podobnou otázku? Při rozhovorech s lidmi zabývajícími se sazbou časopisů je tuto otázku často slyšet. Útržkovité informace a neúplný přehled je málo přesvědčivým důkazem o rozvoji T_EXu v Československu. Rád bych tedy opatřil potřebné informace nejen pro sebe, ale především pro vás a všechny zájemce, kteří váhají s přechodem časopisů na elektronickou sazbu pomocí T_EXu. Proto vás prosím o zaslání odpovědí na následující otázky:

1. Úplný název časopisu (ev. doplněný jeho zkratkou)
2. V jakém jazyku či jazycích jsou příspěvky přijímány.
3. Kdo časopis vydává (včetně adresy redakce).
4. Rozsah časopisu (kolikrát ročně vychází a rozsah stran jednotlivého čísla).
5. Náklad časopisu.
6. Od kterého roku(ročníku) je sázen Tem.
7. Jaké makro je při sazbě používáno.
8. Jaká je technologie výroby.
9. Stručná charakteristika časopisu a poznámky, které jsou podle vás užitečné.

Pro vaši lepší představu uvedu, jak jsem tyto položky vyplnil pro časopis CMUC:

```
\nazev      CMUC - Comentationes Mathematicae Universitatis Carolinae
\jazyk     výhradně anglicky
\vydavatel  Matematický ústav UK, Sokolovska 83,
            186 00 Praha 8 -- Karlín
\rozsah    4xročně, přibližně 200 stran/číslo
\naklad     600 výtisků
\texem     od roku 1989 (ročník 30)
\makro     AmSTeX a upravený styl amsppt.sty
\technologie Tiskové předlohy z-laserové tiskárny v hustotě 300 dpi jsou
            zpracovány dále fotocestou.
\poznamky  Specializovaný matematický časopis.
```

Věřím, že ti čtenáři, kteří mají výše požadované informace o časopisech ze svého okolí, se o ně podělí s ostatními mým prostřednictvím. Je několik důvodů, proč tyto informace shromažďovat centrálně. Jedním z těch důležitějších je to, že podmínkou pro pravidelné zasílání TUGboatů naší organizaci (k zapůjčení v Praze, Brně a Bratislavě — viz T_EXbulletiny č. 2 a 3) je pravidelná zpráva v TUGboatu o T_EXových aktivitách v Československu. Zdá se mi, že seznam časopisů připravovaných T_EXem je jeden z hmatatelných důkazů, že se v Československu něco děje i v oblasti T_EXu. Zároveň to je jistotou propagací vydavatelů časopisů.

Na závěr snad uvedu velmi neúplný seznam časopisů, o kterých se mi výše uvedené informace podařilo zjistit. (Pokud víte, že některé informace nejsou úplně přesné, sdělte mi to laskavě také.)

Časopis: **CMUC – Comentationes Mathematicae Universitatis Carolinae** (Specializovaný matematický časopis. Příspěvky jsou výhradně anglicky).

Vydává: Matematický ústav UK, Sokolovská 83, 186 00 Praha 8 – Karlín;

Doplnění: Použitý formát je $\mathcal{A}\mathcal{M}\mathcal{S}$ -T_EX a upravený styl `amspt.sty` s technologií zpracování: Tiskové předlohy z laserové tiskárny v hustotě 300 dpi jsou zpracovány dále fotocestou. Tem připravován od roku 1989 (ročník 30) a vychází 4× ročně, přibližně 200 stran/číslo v nákladu 600 výtisků.

Časopis: **GARDE** (Příspěvky jsou česky). Vydává: Český šachový svaz

Doplnění: Použitý formát je speciální nadstavba plain T_EXu s technologií zpracování: Tiskové předlohy z laserové tiskárny v hustotě 300 dpi jsou zpracovány dále fotocestou. T_EXem připravován od roku 1991 (ročník 0) a vychází 12*times* ročně v rozsahu 24 nebo 32 stran v nákladu 1200–1500 výtisků.

Časopis: **Differential Geometry and its Application** (Příspěvky jsou anglicky). Vydává: Elsevier Science Publishers B.V. (Sazba v Ústavu matematiky, Slezská univerzita v Opavě, Bezručovo nám. 13, Opava)

Doplnění: Použitý formát je $\mathcal{A}\mathcal{M}\mathcal{S}$ -T_EX (upravený styl `amspt.sty`) a L^AT_EX (upravený styl `article.sty`). Úpravy jsou provedeny tak, aby dávaly shodný výstup. s technologií zpracování: Tiskové předlohy na křídovém papíře z HPLJ IIIP, odesíláno vydavateli. T_EXem připravován od roku 1991 (ročník 1) a vychází 4× ročně, rozsah asi 100 stran/číslo v nákladu ???.

Časopis: **BAJT** (Příspěvky jsou česky).

Vydává: Svaz uživatelů výpočetní techniky

Doplnění: Použitý formát je plain T_EX a L^AT_EX s vlastním stylem s technologií zpracování: Tiskové předlohy z laserové tiskárny v hustotě 300 dpi jsou zpracovány dále fotocestou. Tem připravován od roku 1990 (ročník 1) a vychází 12× ročně, rozsah 80 stran/číslo v nákladu 25.000 výtisků.

(*Oldřich Ulrych*)

e-mail: `ummou@csearn`

Redakční poznámka

Dovolte mi uvést několik praktických problémů, se kterými jsme se setkávali při vydávání jednotlivých čísel \TeX bulletinu.

1. Prakticky každé číslo bylo připravováno ve značné časové tísní (což neumožňovalo důkladnější kontrolu).
2. U příspěvků dodávaných elektronickou poštou (pokud nebyly bezpečně zakódovány) došlo vlivem přenosu k „deformaci“ textu a jejich „dešifrování“ nemuselo být vždy bezpečně provedeno.
3. Rovněž se vyskytl případ, kdy v příspěvku nebyly jednoznačně definovány počáteční podmínky (např. obsahy čítačů používaných při číslování ve vlastních definicích) a při závěrečném zpracování všech článků najednou došlo ke kolizi.

Tedy závěrem se omlouvám autorům a čtenářům za případné nejasnosti v textu, které mohly být způsobeny jedním či více vlivy uvedenými výše. Ne vždy bylo v našich silách a časových možnostech udělat vše tak, jak bychom si přáli. Věřím, že se v příštím roce podaří nejen tyto překážky překonat ale případně i vzhled našeho časopisu vylepšit.

(*Oldřich Ulrych*)
e-mail: `ummou@csearn`

Na následující stránce přinášíme ukázkou dvou písem jako pozvánku na přednášku Yannise Haralambouse, velkého krotitele v písmové aréně. Jeho přednáška se koná



























**14. prosince T991 v 9.³⁰
v Modré posluchárně RUK, Celetná 20**

Vydalo:	Československé sdružení uživatelů \TeX u vlastním nákladem jako interní publikaci
Počet výtisků:	400
Písmo:	Computer Modern
Vytištěno na:	MFF UK v Praze
Adresa:	ČS \TeX UG, MÚ UK, Sokolovská 83, 186 00 Praha 8 – Karlín

Test of yswab on March 30, 2000 at 1654

	'0	'1	'2	'3	'4	'5	'6	'7	
'02x	ı	ı	`	´	˘	˙	˚	˛	"1x
'03x	.		ß						
'04x		!	"	#		%		,	"2x
'05x	()	*	+	,	/	.	/	
'06x	0	1	2	3	4	5	6	7	"3x
'07x	8	9	:		§	=		?	
'10x		Α	Β	Γ	Δ	Ε	Ζ	Θ	"4x
'11x	Ϛ	ϛ	Ϝ	ϝ	Ϟ	ϟ	Ϡ	ϡ	
'12x	Ϣ	ϣ	Ϥ	ϥ	Ϧ	ϧ	Ϩ	ϩ	"5x
'13x	ϫ	Ϭ	ϭ		„]	^	.	
'14x	˙	a	b	c	d	e	f	g	"6x
'15x	h	i	j	k	l	m	n	o	
'16x	p	q	r	s	t	u	v	w	"7x
'17x	x	y	z	—	—	”	~		
'20x		ff	ft	ff	ff	dh	df		"8x
'21x		á	ã			s			
'22x	ı	ı							"9x
'23x		ó	ö				ú	ü	
'24x					§			ß	"Ax
'25x									
	"8	"9	"A	"B	"C	"D	"E	"F	

Test of yinit on March 30, 2000 at 1655

	'0	'1	'2	'3	'4	'5	'6	'7	
'10x									"4x
'11x									
'12x									"5x
'13x									
	"8	"9	"A	"B	"C	"D	"E	"F	