



PUBLIKACE V OPEN SOURCE KVALITNĚ A EFEKTIVNĚ

RYBIČKA, Jiří, (CZ)

Úvod

Zpracování textů se dlouhodobě jeví jako nejrozšířenější úloha zejména na osobních počítačích. Prakticky ze dne na den byly opuštěny tzv. „klasické“ technologie horké sazby z kovových písmen a nahrazeny počítačovými programy, které se snaží s různým úspěchem předstírat, že umí vysázet dokument tak, jak to uměli profesionální sazeči v tiskárně. Šmahem byly vyměněny psací stroje používané více než sto let pro přípravu rukopisů a dokumentů menšího rozsahu a nákladu za počítače pracující s knižním písmem.

Všichni ale víme, že samotný program nestačí – je k tomu potřeba i kvalifikovaný uživatel. A zde nastal zásadní rozpor: nástroj pro sazbu se dostal do rukou prakticky každému, kvalifikace však jen málokomu. Masivně se tedy vytvářejí dokumenty, které nespĺňují základní kritéria sazby, protože v tom lepším případě se pro jejich úpravu používají pravidla pro psací stroje, jejichž obecná znalost a dostupnost je vyšší než znalost a dostupnost pravidel sazby.

Navíc zde hraje roli i efektivnost přípravy: kdo by se „páral“ s nějakými speciálními mezerami, pomlčkami, zarovnávaním a rozpalem? Přece „stačí“ naházet jakkoliv text na stránky – a je to!

Je jasné, že ti uživatelé, kterým není všechno jedno, se zamýšlejí, jak tuto masově používanou aplikaci zvládnout co nejjednodušším a nejlevnějším způsobem. Těm je určeno několik myšlenek tohoto článku.

Kvalita

Kvalitu můžeme chápat jako míru shody s všeobecně uznávanými pravidly a doporučeními. Jedná se o shodu s typografickými pravidly, pravidly pravopisu, normami a případnými

dalšími doporučeními. Klíčovým problémem v této souvislosti je, zda tvůrce dokumentu umí své požadavky formulovat v souladu s uvedenými pravidly a doporučeními. Vlastnosti použitého programového systému mohou v tomto směru pomoci, ale i klást významné překážky.

Chceme-li vytvořit dokument kvalitně a zároveň také efektivně, můžeme u každého uvažovaného programového systému sledovat uvedená kritéria a snažit se dosáhnout u každého z nich co nejpříznivějšího stavu. Lze říci, že každý systém vyžaduje poněkud jiný přístup a hodně záleží na uživateli, co preferuje, co je schopen a ochoten do řešení vložit a jak je zainteresován na výsledku.

Z počítačného pohledu možná vyplývá, že hlavní část kvalitativních parametrů dokumentů je dána především typografickými pravidly. Ne že by například pravidla pravopisu byla druhořadá, ale v dodržování pravopisu nejsou mezi programovými systémy prakticky žádné rozdíly, stejně tak se jen nepatrně podílí zvolený program na dalších kvalitativních nárocích, jako například na způsobu zápisu bibliografických citací apod.

Typografie je tedy z hlediska tvorby dokumentů poměrně zásadní a tento fakt platí v dnešní době nesrovnatelně více než například před 25 lety, protože narozdíl od minulosti dnes upravují dokumenty lidé, kteří neprošli žádnou typografickou přípravou, přestože disponují dokonalými prostředky pro tvorbu dokumentů, o jakých se dřívějším úpravcům a typografům ani nesnilo.

Řešení kvality je z velké většiny závislé nejspíše na tom, kolik toho uživatel o kvalitě (zejména typografické) ví a do jaké míry je ochoten tato pravidla respektovat a ve svých dokumentech také realizovat.

Efektivnost

Efektivnost je spojena s několika aspekty. Co pravděpodobně všichni chceme?

- ušetřit peníze,
- ušetřit čas,
- zjednodušit práci.

Co je potřeba udělat, abychom jednotlivé cíle splnili?

Ušetřit peníze

Používání volně dostupného softwaru primárně šetří peníze a je velmi důležité v sektorech, kde nejsou obvykle k dispozici finanční prostředky pro pořizování softwaru (například ve školství).

Nejpoužívanější volně dostupné programy pro zpracování textů bezesporu jsou OpenOffice Writer, T_EX a jeho nadstavby, Scribus. Tyto programy se velmi liší svým zaměřením, pojetím a všeobecnou znalostí mezi uživateli.

Při používání volně dostupných programů se setkáváme většinou s intuitivně chápanými kritérii, která můžeme shrnout do jedné věty: Uživatel zvolí takový program, v němž je schopen a ochoten pracovat (například preferuje interaktivní ovládání). Obvykle první program, se kterým se seznámí, je ten, ve kterém pak pracuje stále a používá ho na všechny aplikace.

Ušetřit čas

Čas je potřebné počítat ve dvou rovinách:

- čas potřebný k naučení a ovládnutí daného programového systému,
- čas potřebný k realizaci určitého dokumentu v daném systému.

Uvažujeme-li obě části, je zřejmé, že časový nárok na ovládnutí systému je investován jednorázově, zatímco čas věnovaný realizaci dokumentů je nutné vynakládat opakovaně. Optimalizace tedy může spočívat v tom, že ovládnutím většího počtu funkcí nebo změnou přístupu k práci na dokumentu lze ušetřit opakované časové ztráty na jednotlivých dokumentech.

Uživatelé však obvykle nemají dostatek trpělivosti k tomu, aby se důkladně seznámili s nástrojem, který používají. Dalším negativním aspektem jsou neustálé změny verzí a uživatelských rozhraní různých produktů, které vedou k nutnosti zvýšit a opakovaně vynakládat časovou investici věnovanou k ovládnutí systému.

Zde asi ani nerozhoduje, zda pracujeme se systémem volně dostupným nebo komerčním, přístup uživatelů je velmi podobný. Komerční programy se však častěji uchylují ke změnám uživatelského rozhraní, protože je to jeden z nástrojů konkurenčního boje a jeden z prostředků opakovaného získávání licenčních poplatků.

Zjednodušit práci

Efektivnost lze intuitivně pochopit i tak, že dosáhneme požadovaného cíle pomocí menšího počtu méně náročných kroků. To souvisí jednak se schopností přenést těžiště práce na počítač, jednak s optimálně zvolenou strategií postupu celého zpracování dokumentu.

V tomto bodě je zcela zřejmé, že problém spočívá ve vybavenosti uživatele vhodnými informacemi, znalostmi a dovednostmi. Těžko přesvědčíte jakéhokoliv autora, že má ke zpracování dokumentu používat vlastní formátovací styly, když neví, co to je a jak se to používá. Bohužel většina příruček k programovému vybavení neřeší kvalitu a možnosti použití jednotlivých nástrojů, protože jen radí, *jak* něco udělat, ale neradí, *proč* právě tuto službu zvolit.

Stejně nevýhodné je používat určitý programový systém pro tvorbu dokumentu, který se svým charakterem vymyká celému zaměření programu a klade na uživatele nepřiměřené nároky při přípravě. Bez schopnosti pracovat ve více různých systémech však tento uživatel nemá jinou volbu a často ani netuší, kde dělá zásadní chybu.

Dokumenty v systému L^AT_EX kvalitně a efektivně

Uvedli jsme obecně několik aspektů kvality a efektivnosti, soustředíme se nyní na konkrétní řešení. Takový rozbor bychom mohli (a také měli) udělat pro jakýkoliv systém, který pro zpracování textů používáme nebo máme v úmyslu použít. Z toho pak vyplyne jedna velmi důležitá informace – pro jaký druh dokumentů je daný systém optimální, jakých jeho vlastností si máme všimnout a kde lze spatřit nejefektivnější nástroje pro zpracování dokumentu.

Celý postup budeme ilustrovat na systému T_EX, přesněji řečeno na jeho nadstavbě L^AT_EX. V tomto místě nemáme dost prostoru na podrobné seznámení (potřebné informace lze nalézt v četné literatuře), budeme se proto zabývat spíše obecnějšími vlastnostmi.

Kvalita dokumentů a její dosažení

O typografických pravidlech bychom mohli pojednávat na stovkách stránek a o nejrůznějších prvcích včetně jejich možné realizace v počítači byla publikována řada knih a nepřehledné množství elektronických dokumentů. Zaměříme se tedy pouze na některé prvky, jejichž řešení je typické pro systém L^AT_EX.

Písma a speciální znaky, odstavcová sazba

Kvalitní dokument vysázený knižním písmem splňuje nejnáročnější kritéria *čitelnosti*. Ta je podmíněna precizní sazbou, tedy rovnoměrným rozmístěním světla mezi znaky ve slovech, mezi slovy v řádcích, mezi řádky a odstavci stránky. I ve zcela hladkém textu musí být pro dosažení správného výsledku používána řada nejrůznějších rekvizit, počínaje precizní kresbou písma, přes dostatečné informace o znakovém vyrovnání a sliticích až po promyšlenou distribuci mezislovních mezer v odstavci. V tomto směru lze zcela jednoznačně říct, že systém L^AT_EX disponuje pravděpodobně nejpropracovanějším algoritmem odstavcové sazby, který může být řízen řadou parametrů, a může tak vyhovět těm nejvyšším nárokům na preciznost a kvalitu výstupu. Použitá písma v základní distribuci pokrývají běžné potřeby a na rozdíl od jiných systémů nevykazují lokalizační a jiné nedostatky či snad hrubé chyby. Jedno z nejlépe propracovaných písem – Knuthovo Computer Modern – je v moderních distribucích nahrazeno rodinou Latin Modern, obsahující především precizně kreslené akcentované znaky a opravující drobné chyby původních Knuthových vzorů. Rodina obsahuje knižní písmo serifové a bezserifové v běžných řezech a také písmo strojopisné. Specialitou je škálování písma – v žádném jiném systému není v různých velikostech písma používán rozdílný obraz vycházející z optimalizace čitelnosti. Kresba velkých stupňů je tedy odlišná od kresby malých stupňů, písmo této základní rodiny je tedy precizní ve všech aplikacích.

Ve speciálních znacích má systém L^AT_EX rovněž značný náskok před jakýmkoliv jiným programovým systémem. Vysoce kvalitní sazba předpokládá celou škálu rekvizit, kterými systémy založené na bázi T_EXu běžně disponují. Jde o široký repertoár mezerových výplňků (kromě předdefinovaných je možné jakýkoliv myslitelný výplněk vytvořit), dále pomlčky

a správně se chovající spojovníky (zdvojování při řádkovém zlomu), uvozovky v několika variantách a širokou paletu znaků a symbolů používaných v odborných textech.

Stránky a sestava dokumentu

Stránkový design je ovlivňován především samotným sazebním materiálem. Jedná se o přístup, kdy prvotní je velikost materiálu, druhotně se pak odvozuje stránkové rozložení. Na rozdíl od tohoto konceptu je v jiných typografických systémech naopak prvotní stránkové rozložení, do něhož se pak jako do připravených „nádob“ takzvaně „nalévá“ sazební materiál.

Otázkou je, který z obou přístupů je lepší. Lze jednoznačně prohlásit, že tato otázka je závislá pouze na charakteru dokumentu. Pro dokumenty typu kniha, odborná zpráva, vědecký článek v časopisu apod. je zcela bezkonkurenčně lepší přístup \LaTeX ový, naopak pro reklamní leták, jehož tvorba spíše připomíná kreslení v obrazovém editoru, je lepší zmíněný přístup alternativní. V tomto ohledu je vhodné podle charakteru dokumentu zvolit optimální systém pro zpracování.

Řízení stránkového zlomu v systému \LaTeX je doplněno o jedinečný koncept plovoucích objektů, tj. prvků, jejichž finální umístění není dáno uživatelem, ale rozmístovacím algoritmem, který při vhodném nastavení výrazně usnadňuje sesazení celého dokumentu. Nejčastěji se takto obsluhují obrázky a tabulky, patří sem i marginálie (poznámky na okraji). Zcela automaticky pak fungují i poznámky pod čarou, to však není vzhledem k ostatním systémům nic výjimečného.

Koncept sestavy dokumentu uvedeným způsobem velmi kvalitně realizuje vysazení stránek. Lze přesně řídit meziodstavcové mezery, nadpisy a jejich umístění v rámci textu, mezery mezi textem a dalšími objekty, a to vše zcela jednotně, případně s možností určitého akceptovatelného rozmezí. Lze tak dodržet základní typografická pravidla jednotnosti designu a rovnoměrného vyplňování stránek.

Netextové objekty

Pokud jde o matematické výrazy a jejich obdobu ve fyzikálních a jiných odborných textech, lze s jistotou konstatovat, že neexistuje jiný programový systém, který by tuto problematiku řešil kvalitněji a efektivněji. Propracovaný systém matematické sazby s mnoha rekvizitami a přídatnými balíky umožňuje splnit ta nejnáročnější kritéria.

Tvorba tabulek je naproti tomu pravděpodobně nejméně propracovaným prvkem v systému \TeX i \LaTeX . Přestože existuje poměrně velká skupina rekvizit a přídatných balíčků, je v mnoha případech vytvoření tabulky těžkopádné a některé prvky nejsou dostupné vůbec. Na druhou stranu je však potřebné zdůraznit, že pro běžné potřeby tyto rekvizity zcela dostačují a v určitých prvcích lze opět získat kvality, které nejsou v jiných systémech vůbec dostupné, nebo jsou realizovatelné s velkými obtížemi a nároky. Je tedy možné i v oblasti tabulek dosáhnout vysoké typografické kvality.

Obrázky lze do dokumentu v systému \LaTeX včlenit několika způsoby. Většinou se předpokládá, že hotový obrázek v rastrovém formátu JPG nebo PNG, či vektorovém formátu

PDF bude vložen do připraveného vynechaného místa dokumentu, to však obvykle není zařízeno přímou sazbu, ale až výstupním ovladačem. Tento způsob práce nepředstavuje velké omezení, neboť samotná příprava obrazu může být realizována jakýmkoliv vhodným obrazovým editorem, který má pro tuto činnost zcela optimální vybavení. Kromě vložení externě připraveného obrazu existují určité omezené možnosti kreslení vektorového obrazu pomocí jednoduchých prvků, což je často využíváno zejména pro jednoduchá schémata nebo nelineární rozmístění libovolných prvků na plochu stránky.

Efektivnost tvorby dokumentů

Uvažujeme-li o efektivní výrobě dokumentů, určitě pro každý programový systém platí, že její podmínkou je především

1. dokonalé využití existujících rekvizit a předdefinovaného způsobu práce systému.

Efektivnost má v systému \LaTeX oproti jiným systémům ještě ovšem navíc další rovinu –

2. téměř neomezené možnosti rozšíření a modifikace příkazů a přizpůsobení požadované aplikaci.

Tím se diametrálně odlišuje od jiných systémů, které se uživateli jeví jako uzavřené, tedy množina jejich funkcí je předdefinovaná a konečná.

Systém \LaTeX poskytuje uživateli možnost optimálně řešit každý typ dokumentu podle dané situace, přizpůsobit zpracování možnostem vstupu dat a reprezentovat dokument čistě strukturními značkami. Ilustrujme si tyto možnosti na následujících příkladech.

Příklad 1

Z databázového systému získáme soubor `data.csv` a v něm dostáváme tabulku s údaji o zboží, kterou potřebujeme vhodně vysázet. Každý řádek vstupu obsahuje pět údajů (název, počet kusů, nákupní cena bez DPH, prodejní cena bez DPH, sazba DPH), oddělených znakem svislá čára, která je před prvním i za posledním údajem řádku, např.:

```
|Zubní pasta Microsilver      | 2| 130| 167| 20|
```

Takový vstup stačí upravit pro jednodušší následné zpracování automatickým doplněním jednoho příkazu, například:

```
\zbozi|Zubní pasta Microsilver      | 2| 130| 167| 20|
```

K takovému vstupu připravíme soubor:

```

\def\zbozi|#1|#2|#3|#4|#5|{%
    #1 & #2 & #3 & #4 & #5 \\ \hline}
\begin{longtable}[|l|*4{r|}] \hline
Název & Kusů & Nákup & Prodej & DPH \%
    \\ \hline
\endhead
\multicolumn{5}{|r|}{(pokračování na další straně)}
    \\ \hline
\endfoot
\endlastfoot
\input{data.csv}
\end{longtable}

```

Příklad 2

Potřebujeme udělat sbírku příkladů do fyziky. Protože nebudeme vynalézat kolo, posbíráme osvědčené příklady z různých sbírek. Tuto jednotvárnou práci chceme zadat „levné pracovní síle“. Potřebujeme tedy, aby zápis byl co možná nejjednodušší, například takto:

```

\priklad{Vypočtete, kolik metrů je vysoká věž, z~jejíž střechy
spadla taška až na zem za 3 vteřiny.}{45 metrů}

```

Ve vstupním textu je uvedeno pouze to nejnútnejší, aby bylo možné pochopit, o co se jedná. Nejsou zde uvedeny žádné informace o výsledném tvaru. Uvedené značky (příkazy) se nazývají strukturní a strukturní značkování je prvním základním krokem na cestě k efektivnímu dokumentu.

V této fázi stačí tu zmíněnou pracovní sílu pouze instruovat, že každý příklad začne zápisem `\priklad`, že zadání příkladu je v jedné sorce a jeho výsledek je ve druhé sorce. Vůbec nic nemusí vědět o tom, jak bude takový příklad vypadat ve vysázené sbírce, ani nemusí nic vědět o tom, že úplně stejný příklad bude třeba použít pro promítání přes dataprojektor ve výuce nebo bude figurovat jako doplněk k výkladu v učebnici.

Ale i kdybychom sbírku dělali zcela sami a neměli v úmyslu žádné další aplikace, přesto se vyplatí vstup zapisovat stejným způsobem. V každém případě totiž můžeme od sebe oddělit dva procesy – pořizování a úpravy textu, kdy autor uvažuje víceméně pouze o obsahu, a řešení vizuální podoby, kterou může dělat kdokoliv jiný a také neomezeně mnoha způsoby.

Technicky to provedeme doplněním definice makra `\priklad` například takto:

```

\newcounter{cisprikladu}[section]
\def\thecisprikladu{%
    \thesection.\arabic{cisprikladu}}
\def\priklad#1#2{\stepcounter{cisprikladu}

```

```
\par\medskip
\textbf{Příklad \thecispříkladu:} #1 \par
\smallskip \hspace*{\fill}\textit{(#2)}
```

V tomto jednoduchém případě jsme vytvořili čítač pro číslování příkladů, nařídili jsme jeho nulování v každé sekci a jeho číslo společně s číslem sekce vypisujeme na začátku každého příkladu. Příklady začínají tučným slovem „Příklad“ a odpovídajícím číslem, pak následuje text zadání, za příkladem pak vpravo kurzívou v závorce výsledek.

Lze si snadno představit, že v jiné modifikaci úpravy nemusíme výsledky příkladu vůbec chtít vypisovat, můžeme je dokonce nechat vypisovat v úplně jiné části publikace (například souhrnně v poslední kapitole), zadání příkladů můžeme číslovat jiným způsobem (například při tvorbě písemky můžeme mít „číslování“ A, B atd. a každý příklad může být na samostatné stránce bez výsledků) a podobně.

Co si z rozboru odneseme?

Ze stručného seznámení se systémem L^AT_EX by měly vyplynout tyto informace:

- Zaměření a základní princip – systém preferuje design sazebního materiálu před designem stránek (prvotní je tedy text, nikoliv předdefinovaná úprava stránky), takže text postupně vyplňuje a tvoří stránky, nenalévá se do připravených stránkových šablon.
- Kvalitativní kritéria – jsou k dispozici prostředky pro precizní sazbu, nepřeborné množství speciálních znaků, neomezené možnosti mezerování, pravděpodobně nejlepší algoritmus odstavcového zlomu, ideální podpora matematických výrazů.
- Efektivnost – systém je volně dostupný (finanční nároky na pořízení jsou nulové včetně pořízení potřebných podpůrných programů), podporuje v plné míře strukturní značkování, umožňuje automatizovat sazbu, je velmi vhodný pro rozsáhlé dokumenty, vyžaduje investici do zvládnutí tvorby vlastních příkazů.
- Omezení – nemá příliš rozsáhlé možnosti stránkového designu, nepracuje obvykle v interaktivním uživatelském rozhraní (nepracuje s náhledem dokumentu – to však vadí jen úplným začátečníkům), má některá omezení v tvorbě tabulek.

Od této chvíle je tedy jasné, kdy a na jaké dokumenty lze systém s úspěchem použít. Bude tedy například velmi efektivní pro sazbu sborníku (opakované úkony s každým článkem, nepřiliš složitý stránkový design), ale při tvorbě jednostránkového reklamního letáku se neúměrně nadřeme (chybí propracované funkce pro definici stránkového designu, málo textu, málo možností automatizace).

Pohled do budoucna

Na základě uvedeného příkladu jednoho z mnoha různých systémů jsme předvedli, na co všechno by se měl uživatel soustředit, aby se správně rozhodl pro takový programový systém, který optimálně vyhovuje danému typu dokumentu.

Jenže – který uživatel má čas na to, aby takové rozbory prováděl? Je vůbec efektivní se tím zabývat? Systémů je mnoho a neustále se mění – je možné mít neustále přehled?

Přestože znalosti a informace jsou pro efektivní zpracování dokumentů zcela nezbytné, je na všechny tyto otázky nejčastější odpověď „ne“. Proto by bylo ideální, kdyby tyto rozbory někdo kvalifikovaný udělal dopředu a uživatel mohl získaných výsledků rychle využít.

Tuto myšlenku podporuje navíc možnost rozhodování přenechat počítači, který na základě relevantních informací uživateli nabídne, co má pro svůj konkrétní dokument použít a také *jaké funkce* bude potřebovat a kde se s nimi seznámí.

Na Ústavu informatiky PEF Mendelovy univerzity v Brně vzniká rozhodovací systém pro volbu optimálního programového systému. Principem tohoto systému je, že uživatel zvolí druh dokumentu, který má v úmyslu zpracovat, podle potřeby upřesní požadavky na dokument a svoje vlastní preference (například práce v systému Linux) a rozhodovací stroj odpoví, kterým programem je optimálně daný dokument zpracovatelný a vypíše všechny funkce, které budou potřeba. Ke každé použité funkci je k dispozici nápověda, jak tuto funkci v daném programu použít a kde ji najít. Uživatel tedy nemusí dlouze studovat daný systém, dostane ke svému dokumentu odpovídajícího průvodce.

V současné době je tento rozhodovací systém navržen, přistupuje se k jeho realizaci a shromažďují se potřebná data.

Závěr

Volně dostupné programy nabízejí v oblasti zpracování textů široké možnosti. Chceme-li v této aplikační oblasti postupovat tak, abychom dosáhli co nejkvalitnějšího výsledku s co nejmenším úsilím, musíme jednotlivé programové systémy dostatečně důkladně poznat a ohodnotit.

Příkladem pro zpracování dokumentů určitých typů je systém $\text{T}_{\text{E}}\text{X}$ a jeho nadstavby, kde nejrozšířenější je bezesporu $\text{L}_{\text{A}}\text{T}_{\text{E}}\text{X}$. V tomto systému se nacházejí unikátní nástroje a prostředky pro dosažení maximálně kvalitního výstupu, jako každý jiný systém má však i svá omezení.

Efektivní zpracování dokumentů tedy nespočívá v absolutní preferenci jednoho programového řešení, ale spíše ve vhodné kooperaci různě zaměřených a různě vybavených programů, z nichž je potřebné využívat ty nejefektivnější funkce. K tomu je nezbytné se alespoň rámcově seznámit s několika reprezentanty, které dostatečně ilustrují možnosti a omezení celé skupiny podobných programů.

Pro optimální rozhodnutí o zpracovávajícím programovém vybavení vzniká automatizovaný systém, jehož principem je poskytnutí podstatných informací pro řešení konkrétního

dokumentu konkrétního uživatele s danými požadavky a preferencemi.

Literatura

- RYBIČKA, J. *L^AT_EX pro začátečníky*. Brno: Konvoj, 2003.
- RYBIČKA, J. *Typografie a zpracování textů v programu Word* [online]. 2008. [cit. 10. 6. 2010] Dostupné na <http://akela.mendelu.cz/~rybicka/prez/zpract/jaknaword.rtf>
- RYBIČKA, J., TALANDOVÁ, P., PŘICHYSTAL, J. Teoretický model systému pro optimalizovaný proces výběru programového vybavení. *Acta Universitatis Mendeliana Brunensis*, 2010, č. 2 (v tisku)

Kontaktní údaje

Doc. Ing. Jiří Rybička, Dr.,
Ústav informatiky, Provozně ekonomická fakulta,
Mendelova univerzita v Brně, Zemědělská 1, 613 00 Brno, Česká republika,
e-mail: rybicka@mendelu.cz