

Zpravodaj Československého sdružení uživatelů TeXu

Zdeněk Wagner
LaTeXová kuchařka/4

Zpravodaj Československého sdružení uživatelů TeXu, Vol. 8 (1998), No. 2, 105–127

Persistent URL: <http://dml.cz/dmlcz/149817>

Terms of use:

© Československé sdružení uživatelů TeXu, 1998

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



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

V tomto článku se popisuje postup, jak modifikovat vzhled názvů kapitol, podkapitol, apod. Důraz je kladen zvláště na pochopení mechanismu, jak L^AT_EX pracuje s kapitolami. Život je velmi rozmanitý a nelze počítat s tím, že již existuje balík, jenž dokáže zcela vyplnit jakékoliv požadavky nakladatele. Nejsou tedy popisována existující makra, ale je zde poskytnut návod, pomocí něhož lze dosáhnout libovolného vzhledu, ať už programováním příslušných maker od začátku nebo modifikací vhodného balíku.

Jak je v tomto seriálu obvyklé, budeme pokračovat v číslování kapitol. Poslední kapitola v předchozí části má číslo 16.

17. Značkování kapitol

Čitelnost dokumentu vyžaduje, aby začátky kapitol byly nějakým způsobem graficky zvýrazněny. Kapitola může začínat na nové stránce nebo je alespoň vertikálně odsazena od předchozího textu. Název je obvykle sázen jinou velikostí písma, může být v kurzívě nebo tučný, dokonce lze použít jiný typ písma. Pod nadpisem také bývá vertikální mezera. To je pouze krátký přehled hlavních rysů.

Je zřejmé, že všechny uvedené požadavky snadno vyplníme vhodnými makry. Pro přechod na novou stránku použijeme `\clearpage` nebo `\cleardoublepage`. Vertikální mezery udělá `\vspace`, dokonce toto makro vypustí mezery, pokud by název začínal na nové stránce, aniž bychom to vynucovali. Změna písma také nečiní L^AT_EXu žádný problém.

Začínající uživatel může dospět k názoru, že vlastně speciální značky pro kapitoly nepotřebuje. Vždyť lze napsat prostě:

```
\vspace{2cm}
{\large\bfseries\makebox[2em][l]{17.}Značkování kapitol}\par
\vspace{5mm}
\noindent ... začátek prvního odstavce kapitoly
```

Takový přístup ale má řadu nepříjemností. Začněme tím, že nebudou fungovat makra pro křížové odkazy `\label` a `\ref`. Ta totiž potřebují informaci, kterou vytvářejí právě speciální značky pro sazbu názvů kapitol¹. Mohli bychom je sice dopsat ručně, ale tím bychom si přidělali dost práce.

¹Podobně jako Donald E. Knuth se autor tohoto článku také dopouští drobných zjednodušujících lží. Celá pravda bude zjevena v jednom z dalších pokračování.

Problematické je také udržení jednotného vzhledu. Zatvrdí odpůrci strukturního značkování budou trvat na tom, že se nespletou a výše uvedenou sekvencí vždy napíšou stejně. Situace se ale zhorší v případě, kdy nakladatel navržené formátování názvů nepřijme a bude trvat na změně. Nalezení všech názvů a jejich jednotné přepsání není vůbec příjemná práce. Pokud vyhodíme kapitolu nebo naopak dovnitř dokumentu kapitolu přidáme, je nutno vše ručně přechíslovat. Také jsme se ještě nezmínili o tom, že obvykle potřebujeme vytvořit obsah.

Velké nakladatelské domy i někteří jednotlivci si udržují databázi s odkazy do mnoha různých dokumentů. Při překročení určitého rozsahu již nelze takovou databázi udržovat ručně a je nutné, aby dokument prohlédl „indexovací robot“ a příslušné záznamy do databáze automaticky zapsal. Takový robot se ovšem ve špatně strukturovaném textu nevyzná.

Je zřejmé, že používání strukturovaných značek pro názvy kapitol ušetří spoustu práce a nabízí mnoho dalších možností, které běžní \LaTeX isté nevyužívají. Mohou však být nezbytné pro vydavatele, kteří dostávají od autorů zdrojové \LaTeX ové texty.

O výhodách strukturovaného značkování svědčí i praxe z redakce Zpravodaje. Některé příspěvky jsou psány v plain \TeX u. Například Petr Olšák ve svých článcích používá:

```
\def\nadpis #1\par
```

```
{... definice vzhledu názvu...}
```

V redakci je doplněn obvyklý obal `\begin{clanek} ... \end{clanek}` a definice nadpisu je nahrazena příkazem:

```
\def\nadpis #1\par
```

```
{\section{#1}}
```

Tím lze dosáhnout jednotného vzhledu článku s minimálními zásahy do textu. Pokud má příspěvek libovolné strukturované značkování, pak tedy vůbec nevádí, že není napsán s použitím stylu pro Zpravodaj.

18. \LaTeX a SGML, aneb struktura dokumentu

Tato část bude možná někomu připadat nudná a snad se bude čtenářům i zdát, že sem vůbec nepatří. Bude se totiž poměrně dlouho psát o záležitostech, které se \LaTeX u netýkají. Přesto se nakonec souvislosti objasní, a navíc bude předveden způsob, jak by každý autor měl nad svým dokumentem uvažovat.

Řeč bude o SGML. Tato zkratka znamená Standard Generalized Markup Language. Znalci mi jistě odpustí zjednodušení, jichž se v následujícím textu budu dopouštět. Cílem dalších odstavců je pouze ukázání základních rysů SGML a jejich přirovnání k \LaTeX u. Jemný úvod do SGML, vhodný pro začátečníky, lze najít na <http://sable.ox.ac.uk/ota/teip3sg/>, odkazy na další užitečné zdroje jsou na <http://www.icpf.cas.cz/wagner/links.html>.

SGML je jazyk sloužící k definici struktury dokumentu. Každý typ dokumentu obsahuje určité elementy (název, jména autorů, kapitoly, obrázky, tabulky, atp.) a pomocí SGML určíme, které z těchto elementů jsou povinné a kde se různé elementy smí vyskytovat. S dokumentem pak provádíme v zásadě dvě operace: validaci a transformaci. Při validaci ověřujeme, že dokument je syntakticky správný, tedy že všechny povinné elementy jsou přítomny na svých místech a že předepsané vztahy mezi jednotlivými elementy jsou dodrženy. Při transformaci převádíme dokument na jinou formu.

Předvedeme si to zjednodušeně na příkladu tohoto Zpravodaje. Budeme se na něj dívat jako na knihu, přičemž jednotlivé články pro nás budou představovat kapitoly. Číslo, které právě čtete, by v SGML mohlo vypadat například takto:

```
<zpravodaj>
  <chapter title="Písmo s variantní šířkou:
    nová naděje pro naše úzké sloupce?"
    author="Mirka Misáková">
    Text článku...
  </chapter>
  <chapter title="Fonty ve formátu TrueType v pdfTeXu"
    author="Pavel Janík ml.">
    Text článku...
  </chapter>
  <chapter title="Recenze: CE Písma, loga, symboly II"
    author="Antonín Strejc">
    Text článku...
  </chapter>
  <chapter title="Fontanasia" author="Michal Kvasnička">
    Text článku...
  </chapter>
  <chapter title="Konverze dokumentů z WordPerfectu
    do LaTeXu -- WP2LaTeX"
    author="Jaroslav Fojtík">
    Text článku...
  </chapter>
  <chapter title="LaTeXová kuchařka/4" author="Zdeněk Wagner">
    Text článku...
  </chapter>
  <chapter title="Kam za TUGboat-mi?" author="Štefan Porubský">
    Text článku...
  </chapter>
</zpravodaj>
```

Uvedený příklad ukazuje, že názvy jednotlivých elementů se uzavírají do

úhlových závorek < a >. Některé elementy mohou mít různé atributy, např. <chapter> v našem případě má název a jméno autora. Snadno najdeme korespondenci mezi <cosi> ... </cosi> a L^AT_EXovým prostředím \begin{cosi} ... \end{cosi}. Rozdíl jejich pojetí bude předveden později.

Podíváme-li se hlouběji do textu, mohli bychom najít jemnější strukturu. Tento článek by mohl v SGML vypadat např. takto:

```
<chapter title="LaTeXová kuchařka/4" author="Zdeněk Wagner">
  V tomto článku se popisuje postup, jak modifikovat...
  <section title="Značkování kapitol">
    Čitelnost dokumentu vyžaduje, aby...
  </section>
  <section title="LaTeX a SGML, aneb struktura dokumentu">
    Tato část bude možná...
  </section>
  ...
</chapter>
```

Při validaci nás bude především zajímat, zda má každá kapitola název. Také jsme viděli, že Zpravodaj smí obsahovat pouze kapitoly (články). Zkontrolujeme tedy, zda se nevyskytuje text mimo kapitolu. Můžeme ale k definici dokumentu přistupit i jiným způsobem. Protože se nic nesmí vyskytovat mimo kapitoly, lze v definici oznámit, že koncovou značku </chapter> lze vynechat. Programy, které zpracovávají SGML dokumenty, pak před <chapter> chybějící ukončující značku doplní. To je významný rozdíl od L^AT_EXu; zde \end{prostředí} nikdy vynechat nesmíme.

Dosud jsme nemluvili o tom, jak je určen vzhled dokumentu. SGML totiž tuto záležitost neřeší². Převod do vizuální formy je pouze jednou z mnoha transformací, kterou lze s dokumentem provést. Zpravodaj musíme převést na podklady pro tisk a do elektronické formy (PDF). K oběma účelům bychom použili L^AT_EX. Kromě toho lze nějakým indexovacím robotem udržovat databázi, která by umožňovala vyhledávání článků, autorů, případně i klíčových slov ve všech číslech Zpravodaje současně.

Uvedený případ Zpravodaje je velmi jednoduchý a použití SGML se zde zřejmě nevyplatí. Představte si ovšem, že chcete dokument pěkně vytisknout, převést jej do PDF, do Online Book ve formátu pro OS/2, Man Page pro UNIX a prezentovat jej na WWW³. Právě k takovým účelům lze využít SGML. Před transformací si navíc ověříme, zda je náš dokument napsán syntakticky správně.

Existuje řada nástrojů pro převod SGML do L^AT_EXu i opačně, ale o těch zde mluvit nebudeme. Jejich popis by totiž vystačil na několik samostatných

²Úmyslně zde zamčlíme DSSSL (Document Style Semantics and Specification Language), neboť to překračuje rámec tohoto článku.

³Jazyk HTML je aplikací SGML.

článků. Důkazem tohoto tvrzení jsou příspěvky [1–4] přednesené letos v březnu na T_EXNorthEast Conference a publikované v TUGboatu.

Velmi dlouho jsme mluvili obecně, zdánlivě odtažitě, takže je už nejvyšší čas, abychom se dostali k jádru problému. Ukázali jsme si, že SGML definuje strukturu dokumentu a umožňuje jeho validaci. Autor tedy dostane od programu vynadáno, když zapomene název kapitoly a jiné podstatné elementy. L^AT_EX takové schopnosti nemá. Vynadá nám, když místo `\chapter` napíšeme `\capter` nebo když vynecháme `\end{itemize}`. Nebude se ale bouřit, když mu předložíme knihu bez kapitol. Zděsíme se až v případě, kdy napíšeme značku `\section`. První sekce v knize bez kapitol totiž bude mít číslo 0.1!

Vévodkyně z Alenčiných dobrodružství v říši divů [5] řekla: „... and the moral of that is – ‘Be what you would seem to be’ – or, if you’d like it put more simply – ‘Never imagine yourself not to be otherwise than what it might appear to others that what you were or might have been was not otherwise than what you had been would have appeared to them to be otherwise.’“

A jaké poučení z toho plyne pro nás? Každý autor by si měl předem promyslet strukturu dokumentu, který hodlá vytvářet, aby jeho výtvar nedopadl jako zmíněná poznámka Vévodkyně.

V dalším textu se už budeme věnovat výhradně L^AT_EXu, i když se k SGML budeme občas vracet.

19. Typy L^AT_EXových dokumentů

Než začneme s výkladem, připomeneme, že v L^AT_EXové kuchařce/3 [7] byla uveřejněna Tabulka 1 na straně 147. V textu je zmínka, že se k ní ještě budeme vracet. Právě zde ji budeme potřebovat. V době, kdy je tento článek psán, je již zmíněné číslo dostupné v elektronické podobě na WWW serveru sdružení <http://www.cstug.cz/>. Vezměte si ji k ruce, abyste se v následujících odstavcích snáze orientovali.

Setkal jsem se s názorem, že třída REPORT.CLS je šablonou pro všechny zprávy. To ovšem není pravda. Autor tohoto výroku zjevně neprovedl řádně analýzu svého dokumentu a nezamyslel se nad schopnostmi L^AT_EXu.

Tvoříte-li dokument v SGML, musíte definovat typ dokumentu. Někdy lze vhodný DTD⁴ najít, jindy musíme DTD upravit k obrazu svému, a pokud máme smůlu, nenajdeme nikde nic použitelného a vše si musíme udělat sami.

Tvoříme-li dokument v L^AT_EXu, potřebujeme „document class“ a případně „package“ (dříve se jim říkalo „style file“). Často si vystačíme se soubory, které najdeme na CTAN. Někdy je musíme modifikovat nebo si přidáváme vlastní makra a jen zřídka píšeme nové třídy z ničeho.

⁴DTD = Document Type Definition.

Standardní L^AT_EX obsahuje třídy pro nejběžnější typy dokumentů. Leslie Lamport analyzoval obvyklé požadavky na knihy, zprávy, články a dopisy. Podle nich vytvořil třídy BOOK, REPORT, ARTICLE a LETTER. Třída LETTER nemá žádné značky pro kapitoly, neboť v dopisech se takové elementy nevyskytují. Jak zjistíme z Tabulky 1 z minulé části, třídy BOOK a REPORT mají na rozdíl od ARTICLE značku `\chapter`. Kniha se od zprávy liší tím, že se tiskne oboustranně a má jiná živá záhlaví. Řadu rysů lze ovšem změnit. I článek lze tisknout oboustranně, což je například vidět na tomto Zpravodaji. Hlavní soubor totiž začíná

```
\documentclass[twoside]{article}
```

Píšeme-li výzkumnou zprávu, musíme se zamyslet, jaké strukturní elementy budeme využívat. Jestliže v textu budeme mít `\section` a `\subsection`, pak musíme použít třídu ARTICLE. Pokud ale z nějakého důvodu trváme na použití třídy REPORT, musíme kapitoly a podkapitoly značkovat pomocí `\chapter` a `\section`.

Co má ale chudák autor udělat, když napíše celou zprávu, v níž použije `\section` a `\subsection`, a pak je mu vnuceno použití třídy REPORT? Přepisování všech názvů je pracné, ale není třeba zoufat. Stačí následující definice:

```
\let\subsubsection=\subsection
\let\subsection=\section
\let\section=\chapter
```

Příkazy lze rozšířit i na další značky jako `\paragraph`, ale pozor: pořadí příkazů je důležité. Je možný též opačný postup, když v dokumentu máme `\chapter` a jsme nuceni použít ARTICLE. Změní se ovšem nejen místa značek v přiřazeních, ale i pořadí příkazů.

20. Číslování kapitol

Když si napíšete příspěvek s použitím stylu pro Zpravodaj, zjistíte, že `\section` vytvoří nadpis bez čísla. Z běžných příruček víte, že nadpis bez čísla se sází makrem s hvězdičkou, tedy pomocí `\section*`. Zdálo by se tedy, že makro `\section` je zcela přepsáno a očíslování názvů nelze dosáhnout jednoduchou úpravou. Opak je pravdou. Číslování bylo odstraněno jednoduchým L^AT_EXovým trikem.

V minulé části jsme se zabývali tvorbou obsahu. Zmínili jsme se o čítači `tocdepth`, jehož hodnota rozhoduje o tom, které úrovně názvů se v obsahu objeví. Nyní si řekneme, že hodnota čítače `secnumdepth` určuje, které nadpisy budou číslovány. Funguje to zcela analogicky. L^AT_EXová kuchařka vyžaduje číslování až do úrovně `\subsection` včetně. Z Tabulky 1 z minulé části tedy plyne, že musíme použít příkaz:

```
\setcounter{secnumdepth}{2}
```

Chceme-li potlačit číslování kapitol v celém dokumentu, měli bychom používat výhradně čítač `secnumdepth`. Nevýhodou hvězdičkových maker není delší jméno. Kromě toho si můžeme nadefinovat

```
\newcommand\s{\section*}
```

a pak dokonce píšeme mnohem méně znaků. Potíž je však v tom, že hvězdičková makra nezapisují názvy do obsahu a neposkytnou informaci pro živé záhlaví.

20.1. Čítače pro číslování kapitol

Číslo pro každou úroveň kapitoly je udržováno ve stejnojmenném čítači. Lze s ním pracovat pomocí běžných L^AT_EXových příkazů. Můžeme například měnit jejich hodnotu. Tak je to prováděno v tomto seriálu. Na začátku tohoto článku je uveden příkaz:

```
\setcounter{section}{16}
```

Ten zajistí, že první kapitola tohoto dílu bude mít číslo 17. Příkaz `\section` totiž nejprve zvětší hodnotu čítače `section` o jedničku a teprve výslednou hodnotu vytiskne. Přitom též vynuluje čítač `\subsection`. Podíváte-li se do rozvoje maker pro sazbu příslušného nadpisu, zjistíte, že inkrementaci čítače `section` zařídí makro `\refstepcounter`. Nulování `subsection` tam ale nenajdete. To se odehrává jinde. Čítač `subsection` byl totiž deklarován příkazem:

```
\newcounter{subsection}[section]
```

To znamená, že čítač `subsection` bude automaticky vynulován při každé změně hodnoty čítače `section`.

Podobně jsou deklarovány i ostatní čítače. Jedinou výjimkou je čítač `part`, který je nezávislý na ostatních čítačích. Pokud chcete, aby změna čítače `part` způsobila vynulování nejvyššího dalšího čítače, musíte v třídě `ARTICLE` použít povel:

```
\@addtoreset{section}{part}
```

Třídy `BOOK` a `REPORT` vyžadují

```
\@addtoreset{chapter}{part}
```

Nezapomeňte, že uvedený příkaz můžete použít jen v místě, kde má znak `,` kategorie 11 – písmeno (viz např. L^AT_EXová kuchařka/1 [6]).

Někdy se může hodit opačný případ: makro `\@removefromreset`, které by odstranilo jméno čítače ze seznamu těch, které se mají automaticky nulovat. Takové makro vytvořil Johannes Braams [9].

Hodnotu čítače můžete zjistit L^AT_EXovým makrem `\value`. Pokud ale chcete hodnotu získat ve svém složitém makru zejména při změně pořadí expanze pomocí `\expandafter`, je výhodnější použít přímo skutečné jméno čítače. To dostanete předsunutím `\c@` před L^AT_EXové jméno. `\value{section}` tedy poskytne stejný výsledek jako `\c@section`. Trocha opatrnosti je nezbytná u přiřazování. Povel `\setcounter{x}{y}` provede *globální* přiřazení `\global\c@x=y`.

Ke každému čítači existuje též makro pro vytištění jeho hodnoty. Jméno tohoto makra získáte předsazením `\the` před jméno čítače. Nadpis v `\section` se tedy tiskne voláním makra `\thesubsection`.

20.2. Špatný způsob změny formátu číslování

V knize *The L^AT_EX Companion* [8] je uveden příklad, jak špatně a správně modifikovat formát číslování. Zde jej uvedeme jen v mírně změněné podobě.

Zarámování čísla jsme dosáhli definicí:

```
\renewcommand\thesubsection
      {\fbox{\thesection.\arabic{subsection}.}}
```

Všimněte si, že jsme přidali i koncovou tečku, která je v českých dokumentech obvyklá, ale standardní L^AT_EXové třídy ji vynechávají. Vše vypadá dobře až do chvíle, kdy pomocí `\ref` chceme odkázat, že špatný způsob je demonstrován v podkapitole 20.2. Pokud podobný přístup využijete pro kapitoly v třídě BOOK, dočkáte se u čísel rovnic, tabulek a obrázků nemilého překvapení.

20.3 Správný způsob

Podkapitola 20.3 uvádí správný způsob. Je nutno modifikovat jiné makro:

```
\renewcommand\@secntformat[1]{\fbox
      {\csname the#1\endcsname}\enspace}
```

Makru je při volání dodán jeden parametr: jméno příslušného čítače. V našem případě je to `subsection`. Konstrukce s `\csname` (viz [10] nebo [11]) vytvoří při expanzi `\thesubsection`. Makro `\enspace` představuje mezeru o velikosti poloviny čtverčku.

Popsané makro je také správným způsobem, jak přidat za čísla kapitol koncovou tečku. Je to provedeno ve stylu pro Zpravodaj. Příslušná definice zní:

```
\renewcommand\@secntformat[1]{\@nameuse{the#1}.\hspace{.5em}}
```

Místo konstrukce s `\csname` je v ní použito L^AT_EXové makro `\@nameuse`. Koncová tečka je tímto přidána jednotně za názvy všech úrovní.

Podívejte se ještě jednu pozorně na číslo u názvu předcházející podkapitoly. Použili jsme v ní změněnou definici `\thesubsection` a současně platilo výše uvedené makro `\@secntformat`. Proto je tečka i za rámečkem.

21. Změna vzhledu nadpisů

Standardní L^AT_EXové třídy neposkytují příliš mnoho možností a z typografického hlediska nejsou moc dobré. Estetičtějším dojmem působí nadpisy vytvářené třídami z balíku NTGCLASS. Pro jednoduché články a zprávy poskytují tyto třídy postačující řešení, ale pro sazbu krásné literatury jsou zcela nevyhovující.

Knižní tvorba vyžaduje mnohem rozmanitější návrh a nelze počítat s tím, že bychom hotová makra někde našli. Dobrý úvod k problematice změny vzhledu nadpisů najdeme v již zmiňované knize *The L^AT_EX Companion* [8]. Kniha je napsána autory současné implementace L^AT_EXu, takže ji lze považovat za přesný zdroj informací. Bohužel však návod zůstává na povrchu, u vysvětlení základních vnitřních maker L^AT_EXu, ale to v mnoha situacích nestačí. V následujících podkapitolách uvedeme popis základních maker, ale pustíme se hlouběji do nitra L^AT_EXového kernelu, tak jak jsem jej byl kdysi nucen prostudovat. Vysvětlování bude ilustrováno na příkladech skutečných knih.

21.1. Makro `\@startsection`

Toto makro se používá vnitřně pro definice téměř všech příkazů pro názvy kapitol. Výjimkou jsou pouze názvy nejvyšších úrovní, kde je požadován přechod na novou stránku a složitější formátování. Definice makra pro danou úroveň nadpisu pak nemá žádné parametry a při volání `\@startsection` musí dodat šest parametrů. Případnou hvězdičku a nepovinný parametr v hranatých závorkách `\@startsection` ošetří. Syntakticky vypadá volání takto:

```
\@startsection{name}{level}{indent}{beforekip}{afterskip}{styl}
```

Význam jednotlivých parametrů je následující:

name je jméno příkazu, který definujeme, ale bez zpětného lomítka, tedy např. `section`. Z minulého dílu asi tušíte, že jej makro musí znát, aby poslalo správnou informaci pro vytváření obsahu.

level je úroveň nadpisu. Tu musíme zadat nejen kvůli správnému zápisu do obsahu, ale i pro rozhodnutí, zda se má příslušný nadpis číslovat.

indent je odsazení nadpisu od levého okraje. Záporná hodnota způsobí vyčnívání nadpisů vlevo mimo sazební obrazec.

beforekip je vertikální mezera mezi nadpisem a předcházejícím textem. Je to rozměr typu *glue*, takže smí obsahovat `plus` a `minus`. Pokud je zadaná hodnota záporná, bude první odstavec vysazen bez odstavcové zářádky. Skutečná vertikální mezera je vždy určena absolutní hodnotou. Protože před sazbou nadpisu L^AT_EX zahájí odstavec, přidá se k zadané hodnotě ještě `\parskip`.

afterskip je vertikální mezera pod nadpisem. Pokud je ale zadaná hodnota záporná, pak začíná text na stejném řádku a za nadpisem se vynechá místo určené záporně vzatou hodnotou (tedy kladné). Zadává se opět rozměr typu *glue*. Před první odstavcem T_EX přidá `\parskip`. Z toho vyplývá, že při použití makra `\@startsection` nelze pod nadpisem vynechat menší prostor než `\parskip`.

style definuje styl nadpisu. Lze použít libovolné příkazy, které mění styl písma a sazby. Nadpis je sazen uvnitř skupiny, takže zadané příkazy nemohou

ovlivnit vzhled následujícího textu. Můžeme použít například `\bfseries`, `\Large`, `\centering` a mnoho jiných.

Jako jednoduchý příklad uvedeme definice ze stylu pro tento Zpravodaj. Zde používáme jen `\section` a `\subsection`.

```
\renewcommand\section
  {\@startsection{section}{0}{0pt}%
  {-22dd minus -6pt}{11dd minus 3pt}{\raggedright\large\bfseries}}
\renewcommand\subsection
  {\@startsection{subsection}{1}{0pt}%
  {-22dd minus -6pt}{1pt minus 1pt}{\raggedright\bfseries}}
```

Všimněte si, že v rozměrech typu *glue* v uvedených definicích není část plus. \TeX totiž každou roztažitelnost považuje principiálně za nekonečnou. Za jistých okolností se \TeX může rozhodnout, že na stránce není dostatečně vhodné místo pro zlom, oznámí `Underful \vbox` a roztažitelné mezery rozpálí. Tím v sazbě mohou vzniknout obrovské díry. Naproti tomu je stažení vždy konečné. V nejhorším případě pak \TeX nechá větší mezeru dole, což nebývá tak ošklivé, jako nadměrné mezery uprostřed stránky. Také *afterskip* u `\subsection` je deklarován jako `1pt minus 1pt`, což je velmi malá kladná hodnota a na rozhození řádkového rejstříku se téměř neprojeví. Ještě menší kladnou hodnotou je `1sp minus 1sp`, ale vizuálně v tom nebude velký rozdíl.

21.2. Makro `\@afterheading`

Nadpisem kapitoly nesmí končit stránka, ale musí být pod ním nejméně dva řádky prvního odstavce. Pokud by na stránce nebylo dost místa, musíme vysadit nadpis na novou stránku. To vše lze snadno vyřešit vhodnými penaltami. \LaTeX má vše potřebné definováno v makru `\@afterheading`, jež je posledním povelům všech příkazů pro sazbu nadpisů kapitol. Makro ignoruje veškerý vertikální materiál. Před sazbou prvního odstavce zkontroluje logický přepínač `\if@afterindent`. Podle něj se rozhodne, zda má první odstavec mít odstavcovou zarážku. Na konci odstavce obnoví původní hodnoty penalt.

Stejným způsobem funguje vnitřně i `\@startsection`. Podle znaménka hodnoty *beforeskip* se nastaví `\@afterindenttrue` nebo `\afterindentfalse` a zavolá se `\@afterheading`.

21.3. Makra `\secdef` a `\@dblarg`

Nadpisová makra mají tři varianty: základní, s nepovinným parametrem v hranatých závorkách a s hvězdičkou. Makro `\@startsection` tyto případy řeší uvnitř, nemusíme se tudíž o nic starat. Pokud však vytváříme složitější nadpisové makro, musíme si hvězdičku i hranaté závorky ošetřit sami. Autoři \LaTeX u nám pomohli tím, že v kernelu dali k dispozici makro `\secdef`.

Makro `\secdef` má dva parametry. Těmi jsou příkazy, které se mají vykonat podle toho, zda následuje či chybí hvězdička. Například makro `\part` lze definovat způsobem:

```
\renewcommand\part{ ... \secdef \normalpart \starredpart}
\newcommand\normalpart[2][?]{ ... }
\newcommand\starredpart[1]{ ... }
```

Definice makra `\starredpart` odpovídá běžným L^AT_EXovým zvyklostem. Hvězdičku spolklo při své expanzi `\secdef`, takže se o ni nemusíme starat. Složitější situace nastává u `\normalpart`. Všimněte si, že toto makro má nepovinný parametr. Víme, že nepovinný parametr slouží k zadání kratší varianty názvu, která se zapisuje do obsahu a objevuje se v živém záhlaví. Standardní hodnotou tohoto nepovinného parametru je otazník. Přesto můžeme nepovinný parametr vynechat a v obsahu budeme mít správný název. Je to způsobeno tím, že se `\normalpart` zavolá prostřednictvím `\@dblarg`.

Makro `\@dblarg` se používá právě v případech `\macro[arg0]{arg1}`, přičemž standardně má platit `arg0 = arg1`. Pokud za makrem není hranatá závorka, `\@dblarg` vytvoří správný nepovinný parametr. Příkaz

```
\@dblarg{\macro}{arg1}
```

pak bude expandován do tvaru

```
\macro[arg1]{arg1}
```

Je tedy zřejmé, že v případě `\normalpart` se nikdy nemůže otazník do nepovinného parametru dostat bez toho, že bychom jej explicitně uvedli.

21.4. Makro `\refstepcounter`

Často potřebujeme kapitoly číslovat a odvolávat se na ně pomocí makra `\ref`. Již jsme uvedli, že každá úroveň nadpisu má přidělen svůj čítač. Hodnotu tohoto čítače musíme před vytištěním zvýšit o jedničku. Šlo by to provést různými způsoby, ale musíme použít jediný: makro `\refstepcounter`. To kromě zvýšení hodnoty čítače ještě vytvoří další pomocná makra, která budou následně použita makrem `\label`. Pokud bychom použili `\stepcounter` nebo `\addtocounter`, nefungovaly by křížové odkazy.

Za výjimečných okolností můžeme vyžadovat nastavení čítače na hodnotu, která není o jedničku vyšší než předchozí. Pokud mají i v tomto případě fungovat křížové odkazy, musíme příkazem `\setcounter` vložit hodnotu o jedničku nižší a následně použít `\refstepcounter`.

21.5. Makra `\markboth`, `\markright` a přátelé

V odborných knihách máme často živé záhlaví, které obsahuje název aktuální kapitoly a podkapitoly, případně můžeme uvádět název knihy. T_EX pro tyto účely

zavádí primitiv `\mark`. Ten je v \LaTeX u rozšířen tak, aby umožnil vložení dvou značek, levé (`\leftmark`) a pravé (`\rightmark`). Ty reprezentují živá záhlaví na levé a pravé části dvoustrany při oboustranném tisku.

Príslušné značky vytváříme pomocí `\markboth` a `\markright`. První z nich má dva parametry a vytváří obě značky. Druhé makro má jediný parametr. Mění pouze hodnotu pravé značku, levou značku ponechá beze změny.

Nyní se vrátíme na chvílku zpět k definici `@startsection`. Prvním parametrem tohoto makra je jméno příkazu pro nadpis, který vytváříme. Toto jméno se použije též při vytváření příslušné značky. Při expanzi pak `\section` volá `\sectionmark`, `\paragraph` zavolá `\paragraphmark` apod. Všechna tato makra mají jeden parametr a standardně jsou ekvivalentní makru `@gobble`, které tento parametr pouze spolkně. Je tedy ponecháno na uživateli, aby tato makra případně předefinoval s použitím `\markboth` nebo `\markright`.

V jednom z dalších pokračování se zaměříme na živá záhlaví. Tam si pak ukážeme, jak se informace z `\leftmark` a `\rightmark` využijí.

22. Příklady

Již jsme uvedli příklad maker pro nadpisy v tomto Zpravodaji. Nyní si předvedeme několik ukázek z různých knih. Některé z nich byly sázeny ještě ve starém \LaTeX u 2.09, případně byly pro právě přicházející \LaTeX 2_ε upraveny „za pochodu“. Proto jsou některá makra programována jinak, než bylo před chvílí teoreticky vysvětleno. Nebudeme zpětně zasahovat do již odladěných maker, pouze na některých místech upozorníme, jak to jde nyní udělat pohodlněji.

Zde publikované ukázky nejsou přesnou kopií citovaných knih. Byly vysazeny znovu s použitím jiných fontů a do změněného zrcadla, aby se vešly do formátu Zpravodaje.

22.1. Jednoduché nadpisy s živým záhlavím

Začneme s knihou Pataňdžaliho jógasútra [12]. Ta má pouze jednu úroveň nadpisů a pro ně bylo zvoleno makro `\chapter`. Hlavní kapitoly jsou číslovány, některé čísla nemají. Pro ty zvolíme variantu s hvězdičkou. Všechny kapitoly se ale musí objevit v obsahu.

Číslovaná kapitola je uvozena slovním označením, např. KAPITOLA PRVNÍ, její název je na dalším řádku. Text je vždy centrováný. Nečíslovaná kapitola má pouze název. Živé záhlaví má na levé straně označení kapitoly (např. KAPITOLA I), vpravo je název. Nečíslované kapitoly mají vlevo název knihy. Podobný případ nastává u poslední stránky kapitoly, pokud se objeví na levé straně. Číslo kapitoly by zde nevypadalo dobře, proto také vysadíme název knihy. Počáteční stránka kapitoly žádné záhlaví nemá.

V makru použijeme nejprve `\secdef`.

```
\def\chapter{\par
  \ifodd\c@page \else
    \markboth{Pataňďžaliho jógasútra}{\rightmark}
  \fi
  \clearpage
  \thispagestyle{empty}\global\@topnum\z@
  \@afterindentfalse
  \secdef\@chapter\@schapter}
```

Všimněte si, jak ošetříme změnu živého záhlaví. Odstavcovou zarážku zrušíme makrem `\@afterindentfalse`. Vnitřní čítač `\@topnum` zakáže plovoucí obrázky a tabulky na horním okraji stránky, aby se nemohly objevit nad nadpisem. V této knize ale žádné plovoucí objekty nejsou, takže příkaz byl vlastně zbytečný.

Hvězdičková verze musí vysadit název kapitoly a zapsat informaci pro obsah. Musí též nastavit značky pro živé záhlaví.

```
\def\@makeschapterhead#1{%
  \vspace*{3ex}%
  {\centering\sc\Large
  #1\par}\nobreak\vskip 4ex minus 7\p@}
```

```
\def\@schapter#1{%
  \edef\scratch{#1}%
  \ifx\scratch\contentsname \else
    \addcontentsline{toc}{schapter}{\uppercase{#1}}\fi
  \chaptermark{#1}\@makeschapterhead{#1}\@afterheading}
```

V minulé části L^AT_EXové kuchařky [7] jsme si řekli, že obsah je obvykle uvozen příkazem `\chapter*{\contentsname}`. Vypadalo by komicky, kdyby obsah začínal odkazem na sebe. Pokud je tedy názvem kapitoly `\contentsname`, žádná informace pro tvorbu obsahu se nezapíše.

Knihla začíná obsahem, což je technicky nečíslovaná kapitola, a nečíslovanou předmluvou. Další kapitoly jsou číslovány. Můžeme tudíž `\chaptermark` definovat jednoduchým způsobem.

```
\def\chaptermark#1{\ifnum\c@chapter=\z@ \markright{#1}\else
  \markboth{Kapitola \Roman{chapter}}{#1}\fi}
```

Standardní verze nadpisového makra se liší jen v detailech. Všimněte si, že pro jeho definici je použit způsob z plain T_EXu. Makro `\@chapapp`, které se v L^AT_EXovém kernelu dost používá, expanduje na `\chaptername`, což je v českém

prostředí definováno jako „Kapitola“. Zde jsme jej předefinovali na tentýž text psaný verzálkami.

```
\def\@makechapterhead#1{%
  \vspace*{3ex}%
  {\centering\sc\@chapapp{}} \thechapter\par
  \vskip 1ex
  \Large#1\par}\nobreak\vskip 4ex minus 7\p@}

\def\@chapter[#1]#2{\refstepcounter{chapter}%
  \addcontentsline{toc}{chapter}{\protect
    \numberline{\Roman{chapter}}{\uppercase{#1}}}%
  \chaptermark{#1}\@makechapterhead{#2}\@afterheading}
```

Zbývá už jen definice čítače. Potřebujeme převedení čísla na slovní vyjádření. Použijeme k tomu primitiv `\ifcase`.

```
\def\thechapter{\ifcase\c@chapter \or PRVNÍ\or DRUHÁ\or TŘETÍ\or
  ČTVRTÁ\or PÁTÁ\or ŠESTÁ\or SEDMÁ\or OSMÁ\else
  \@latexerr{Too many chapters}\fi}
```

22.2. Netradiční vzhled kapitol

Nyní se zastavíme u knihy Milaräpa, velký tibetský jógin [13]. Věnovali jsme se jí i v minulé části pojednávající o tvorbě obsahu. Kniha obsahuje tři úrovně nadpisů, `\part`, `\chapter` a `\section`. Právě `\chapter` je pro nás nejzajímavější, další typy nadpisů zde pomineme.

V knize máme dva typy kapitol. Nečíslované kapitoly mají pouze centrovný název, což nečiní žádné problémy. Číslované kapitoly jsou uvozeny slovem Kapitola a římským číslem, pod nímž je uveden název. Dále je kurzívou vysázen stručný obsah a po vynechaném řádku následuje text, přičemž první slovo je vysázeno kapitálkami. Příklad jedné takové kapitoly vidíte na obrázku 1.

Makro pro tisk hodnoty čítače kapitol je konvenční a nevyžaduje podrobnější vysvětlení.

```
\def\thechapter{\Roman{chapter}}
```

Dále si nadefinujeme logický přepínač a token registr. Budeme je brzy potřebovat.

```
\newif\ifnewpage \newpagetrue
\newtoks\chap@toc
```

Makra pro sazbu nadpisů budou mít neobvyklý tvar, proto zde nelze použít `\secdef`. Přítomnost hvězdičky budeme zjišťovat makrem `\@ifstar`. To má dva

KAPITOLA II CHUŤ STAROSTÍ

Smrt a poslední vůle Milaräpova otce; protiprávní přisvojení si majetku strýcem a tetou z otcovy strany; z toho vzniklé starosti, jež musel Milaräpa snášet se svou matkou a sestrou.

OPĚT pozvedl Rächung svůj hlas: „Důstojný učiteli, pověz nám milostivě o svých útrapách a nesnázích po smrti svého otce.“

Obrázek 1: Příklad nadpisu kapitoly z knihy Milaräpa, velký tibetský jógin.

parametry. První se použije v případě, že následujícím znakem je hvězdička, v opačném případě se použije druhý parametr.

```
\def\chapter{\ifhmode \par \fi \@ifstar{\s@chapter}{\n@chapter}}
```

V knize chceme dosáhnout jednotného vzhledu nadpisů. Proto `\s@chapter` ani `\n@chapter` nic netisknou, pouze připraví údaje pro další příkazy.

```
\def\s@chapter#1{%  
  \def\chap@head{#1}  
  \chap@toc{#1}\zw@chapter}
```

```
\def\n@chapter#1{%  
  \refstepcounter{chapter}  
  \def\chap@head{KAPITOLA \thechapter\|#1}  
  \chap@toc{\numberline{\thechapter}#1}\zw@chapter}
```

Veškerou práci provádí dosti složitě makro `\zw@chapter`.

```
\newcommand\zw@chapter[1][\relax]{%  
  \ifnewpage \clearpage \fi \newpagetrue  
  \def\zw@{#1}\def\zw@next{\relax}  
  \ifx\zw@\zw@next  
    \let\zw@\relax  
    \edef\zw@next{\noexpand\addcontentsline{toc}{chapter}%  
      {\the\chap@toc}}  
  \else  
    \edef\zw@next{\noexpand\addcontentsline{toc}{chapter}%
```



```

    {\noexpand\protect\noexpand\cpt@entry{\the\chap@toc}{#1}}
\fi
\zw@next
\addvspace{45pt minus 7pt}
\begin{center}
  \Large\bfseries\chap@head\par
\end{center}
\nobreak\vskip 28dd minus 7pt
\ifx\zw@\relax \else
  \begingroup \interlinepenalty\@M
  \noindent \itshape#1\par
  \nobreak \vskip\baselineskip
  \endgroup
\fi
\@afterindenttrue\@afterheading}

```

Kapitola má začínat na nové stránce s výjimkou první kapitoly každé části. To se testuje logickým přepínačem `\ifnewpage`. Makro `\part` nastaví tento přepínač na `\newpagefalse`, ve všech ostatních případech máme `\newpagetrue`.

Nečíslované kapitoly zapisujeme pomocí `\chapter*{Nadpis}`, formát číslovaných kapitol je složitější. Stručný obsah uvádíme za názvem v hranatých závorkách, tedy `\chapter{Název}[Text.]`. Stručný obsah se tedy dostane do makra `\zw@chapter` jako nepovinný parametr. Hodnota tohoto parametru v případě nečíslované kapitoly bude `\relax`. Podle toho vložíme do `\zw@next` příslušný příkaz pro zápis informací do obsahu. Část jsme již předem schovali v token registru `\chap@toc`. Ten se expanduje právě jednou, proto před `\numberline` nepotřebujeme `\protect`. Některá makra však při definici expandovat nesmíme. Proto před ně vkládáme `\noexpand`.

Do makra `\chap@head` jsme již předem vložili text nadpisu. Nyní jej pouze obložíme formátovacími instrukcemi. Mezera nad nadpisem je zde definována pro případ první kapitoly v každé části. Ve všech ostatních případech vyjde na začátek stránky a \TeX ji pak ignoruje. V číslovaných kapitolách pak vysázíme obsah nepovinného parametru a skončíme magickou sekvencí `\@afterindenttrue \@afterheading`. Přepnutí prvního slova do kapitálek bylo řešeno ručně. Pokud by totiž věta začínala předložkou, měly bychom raději vytisknout kapitálkami dvě slova. To však nelze automaticky zajistit.

Makro `\cpt@entry` bylo popsáno v minulé části [7]. Zde se jím nebudeme zabývat.

22.3. Vložení obrázku

V knize *Châteaux et Légendes, Promenades au Royaume de Bohême et de Moravie* [14] chceme každou kapitolu ozdobit ilustrací. Příklad je na obrázku 2.



VYŠEHRAĐ

LE CHATEAU AU-DESSUS DE LA VLTAVA

Autrefois, dans les temps anciens, sur la Vltava, se tenait un château avec des fortifications et un palais construits tout en bois. Plus tard, il avait reçu le nom de Vyšehrad. Il se dressait sur la rive droite de la rivière, en haut d'un piton rocheux, baigné par les flots agités. Le château était résistant, comme la volonté de ceux qui y gouvernaient.

Obrázek 2: Příklad nadpisu z knihy Châteaux et Légendes, Promenades au Royaume de Bohême et de Moravie.

V tomto případě bylo nutno vyřešit několik problémů. Ve francouzštině se nepíší akcenty nad verzálkami. Protože názvy kapitol budeme sázet verzálkami, ale v obsahu použijeme minusky, musíme místo `\uppercase` použít jiné makro. Mezi francouzskými slovy se ovšem vyskytnou i česká jména. Nakladatel si přál, aby akcenty u českých jmen zůstaly zachovány. Tyto požadavky byly ošetřeny následujícími makry.

```
\def\save@accents{\let\cz@hacek\v \let\cz@acute}'}
```

```

\def\CZ{\protect\CZ@accents}
\def\CZ@accents#1{\begingroup \let\v\cz@hacek
                  \let\'\cz@acute #1\endgroup}
\def\remove@accents{\let\'\empty \let\'\empty
                   \let\^\empty \let\v\empty}
\def\protect@accents{%
  \def\'\{\protect\'}%
  \def\`{\protect\`}%
  \def\^{\protect\^}%
  \def\v{\protect\v}}

\save@accents

\def\Uppercase#1{\begingroup \let\zw@protect\protect
  \let\protect\noexpand
  {\protect@accents
  \xdef\zw@temp{#1}}\let\protect\zw@protect
  \expandafter\remove@accents\expandafter\uppercase
  \expandafter{\zw@temp}\endgroup}

```

V názvu kapitoly jsme uvedli `\CZ{Vyšehrad}`, čímž jsme obnovili původní význam maker pro diakritická znaménka. Vše funguje pouze proto, že $\text{T}_{\text{E}}\text{X}$ od nás dostává sedmibitový vstup s řídicími sekvencemi. Ve zmíněné knize toho bylo dosaženo použitím TCP tabulky, která mapuje příslušným způsobem osmibitový vstup. Zpravodaj však z vážných důvodů tuto metodu použít nemůže. Při sazbě ukázky jsme museli česká písmena dočasně udělat aktivními. Použili jsme k tomu neobratně napsané příkazy (ale bylo to rychlejší než psaní maker jako v balíku `INPUTENC`):

```

\def\TiFont#1{\catcode'#1\active \def}

\TiFont áá{\'a}
\TiFont ää{"a}
\TiFont čč{\v c}
\TiFont ďď{\v d}
...

```

Hlavní kapitoly sázíme pomocí makra `\hrad`, které má dva parametry. První z nich je definován jako nepovinný. Obsahuje označení obrázku, který se má vysázet. Protože text byl k dispozici dříve než obrázky (u nich byly pouze známy rozměry), byl zvolen právě tento přístup. Označení obrázků jsme nemuseli hned zadávat a bylo doplněno později.

```

\newcommand\hrad[2][blbost]{\clearpage \ZW@nbreaktrue
\chapterhead{#1}%
{\usefont{T1}{Antique}{m}{n}
\fontsize{17}{25pt}\selectfont
\leftskip \z@ plus 1fil \rightskip\leftskip
\parfillskip\z@ \noindent
\Uppercase{#2}\par}\vskip 6ex plus 2ex minus 2ex
\addcontentsline{toc}{hrad}{#2}
\@afterindentfalse \@afterheading}

```

Všimněte si, že nastavujeme logický přepínač `\ZW@nbreaktrue` podobně jako v minulém příkladě. Testujeme jej pak v nadpisu úrovně `\section`.

```

\def\section#1{\ifhmode \par \fi
\ifZW@nbreak \ZW@nbreakfalse \else \clearpage \fi
{\bfseries\leftskip \z@ plus 1fil \rightskip\leftskip
\parindent\z@ \parfillskip\z@
\Uppercase{#1}\par}\vskip 4ex plus 2ex minus 2ex
\addcontentsline{toc}{section}{#1}
\@afterindentfalse \@afterheading}

```

V obou typech nadpisů nastavujeme přímo rozměry `\leftskip`, `\rightskip`, `\parindent` a `\parfillskip`. Dosáhneme tím vycentrování nadpisů. Zde se nabízí otázka, proč nepoužijeme jednoduše prostředí `center`. Je to dáno tím, že v knize máme změněny některé rozměry, a proto standardní makra občas nefungují.

Usazení obrázků provádí makro `\chapterhead`. Jeho definice je trochu komplikovaná.

```

\newcommand{\chapterhead}[1]{%
\edef\cat@colon{\the\catcode '\:}\catcode '\:=12
\def\get@shift##1,##2,##3:%
\def\@@@lft{##1mm}\def\@@@fn{H:/EPSfiles/Martin/m##3.eps}%
\up@shft34.5mm \advance\up@shft##2mm\relax}
\ifZWdraft
\vbox to 34.5mm{\vss\offinterlineskip\hrule
\@@line{\vrule height 34mm\hss\vrule height 34mm}
\hrule}
\else
\makeatletter
\InputIfFileExists{H:/EPSfiles/Martin/m#1eps.tex}{\typeout{***** Cannot find picture: m#1.eps *****}}
\get@shift0,0,0:}

```

```

\makeatother
\ vbox to 34.5mm{\vss
\iftrue
  \moveleft \@@@lft\vbox to \up@shft
                {\noindent\includegraphics{\@@@fn}\vss}%
\fi}
\fi
\catcode ‘\:\cat@colon
\vskip 20.5mm plus 1.8ex minus 1.8ex % only this can have "plus"
}

```

Nejprve si musíme uvědomit, že sázíme francouzský text. V něm jsou některá interpunkční znaménka aktivní. Dvojtečku však potřebujeme v názvech souborů a musí mít kategorii 12. Původní kategorii (víme, že je to 13, ale napíšeme to obecněji) si tedy schováme do makra `\cat@colon`, abychom ji nakonec mohli obnovit.

Obrázky byly skenovány programem, který nedával těsný bounding box. Stejná kniha pak byla sázena v dalších jazykových mutacích. Abychom nedělali stejnou práci dvakrát, byl ke každému obrázku vytvořen jednořádkový soubor s makrem `\get@shift`, z něhož zjistíme korekční posunutí a skutečné jméno souboru s obrázkem. Makro `\get@shift` je definováno uvnitř `\chapterhead`.

Obrázky se sázejí příkazem `\includegraphics`. Pokud v `\documentclass` uvedeme `DRAFT`, pak místo obrázku dostaneme jen obdélník se jménem souboru. Podobná funkce je naprogramována pomocí přepínače `\ifZWdraft`. Jedná se o relikv z pradávých dob. Kód byl totiž opsán ze stylu pro sazbu velmi staré knihy, kdy ještě nebyl $\text{\LaTeX} 2_{\epsilon}$, obrázky se vkládaly makrem `\epsffile` a pracoval jsem v DOSu. Musel jsem tehdy obcházet neexistenci pohodlnějších nástrojů.

23. Křížové odkazy

O křížových odkazech jsme se již zmínili, když jsme si v podkapitole 20.3 ukazovali, jak správně modifikovat formát tisku čítače. Další poznatek jsme uvedli v podkapitole 21.4. Nyní si řekneme trochu více o tom, jak a proč odkazy fungují.

Nejprve bychom se měli zeptat, jak je vůbec možné, že \LaTeX ví, zda se odkazujeme na `\chapter`, `\section` nebo `\paragraph`. Odpověď je velmi jednoduchá. Zařídí to makro `\refstepcounter`. Uloží totiž jméno čítače do proměnné `\@currentlabel`. Makro `\label` podle toho pozná, na který čítač se odkazujeme, a запиše příslušnou informaci do pomocného souboru. Základní \LaTeX ové příručky uvádějí, že nesmíte nadpis kapitoly či podkapitoly vložit do skupiny bez textu. Má to jednoduchou příčinu. Po uzavření skupiny přestane platit na-

stavení proměnné `\@currentlabel` a obnoví se předchozí hodnota. Pokud pak použijete `\label`, budete se odkazovat na předchozí kapitolu.

L^AT_EX umí také vytvořit odkazy na obrázky, tabulky, rovnice i výčty. S plovoucími objekty však má řada uživatelů obtíže. Křížový odkaz jim směřuje na kapitolu, nikoliv na obrázek. Vysvětlení je prosté. Aby `\label` fungoval správně, musíme nejprve naplnit `\@currentlabel`. To provede, jak již víme, makro `\refstepcounter`, které je ale obsaženo v `\caption`. Proto musíme napsat `\label` až za `\caption`. Protože prostředí je vlastně skupina a po jejím uzavření se obnoví stará hodnota `\@currentlabel`, musí být `\label` uvnitř prostředí.

Představme si, že máme čítač pojmenovaný `CNT` a za určitých okolností provedeme `\refstepcounter{CNT}`. Když pak nadefinujeme `\label`, zapíše se do pomocného souboru hodnota získaná vyhodnocením výrazu:

```
\csname p@CNT\endcsname\theCNT
```

O makru `\theCNT` jsme se již zmiňovali. To definuje tiskovou podobu. Makro `\p@CNT` určuje prefix pro křížové odkazy. V makru `\label` se volá pomocí konstrukce `s \csname`. Ono to ani jinak nejde, ale má to navíc tu výhodu, že makro `\p@CNT` nemusíme definovat. Pokud není definováno, expanduje se příslušná konstrukce `s \csname` na `\relax`, což nám žádnou škodu nezpůsobí. Obvykle nejsou makra typu `\p@CNT` definována. Jiná je situace u výčtů (zde se `\refstepcounter` volá při expanzi `\item`). Například v třídě `ARTICLE` se definuje:

```
\renewcommand\p@enumii{\theenumi}  
\renewcommand\p@enumiii{\theenumi(\theenumii)}  
\renewcommand\p@enumiv{\p@enumiii\theenumiii}
```

24. Popisy obrázků a tabulek

Popis obrázků i tabulek vytváříme stejným makrem. L^AT_EX sám pozná, k jakému objektu `\caption` patří. Postarají se o to makra příslušného plovoucího prostředí. Jeho typ se nejprve vloží do `\@capttype`. Makro `\caption` pak provede `\refstepcounter` a zavolá `\@caption` pomocí známého `\@dblarg`. To zapíše informaci do seznamu příslušných plovoucích objektů a vyvolá `\@makecaption` s dvěma parametry: jménem makra pro tisk čísla (např. `\thefigure`) a popisem.

Někdy chceme neplovoucí objekt, ale potřebujeme `\caption`. Lze to jednoduše obejít. Vezměme si příkazy:

```
\makeatletter  
\newenvironment{mytable}{\ifhmode \par \fi  
  \edef\@capttype{table}}{\ifhmode \par \fi}  
\makeatother  
\begin{mytable}  
\caption{Stručný přehled zvířat}
```

```
\begin{center}
\begin{tabular}{|l|l|}\hline
```

```
...
```

Jejich výsledkem bude:

Tabulka 1: Stručný přehled zvířat

Zvíře	Identifikace
Pes	Baskervillský
Kocour	V botách

25. Literatura

Následující přehled obsahuje jak odkazy na zdroje citované v tomto článku a texty vhodné k dalšímu studiu, tak i knihy, z nichž byly převzaty ukázky vzhledu nadpisů kapitol.

1. Robert S. Sutor and Samuel S. Dooley: *TEX and L^AT_EX on the Web via IBM techexplorer*. TUGboat **19**(2), 157–161(1998).
2. Ross Moore: *Presenting mathematics and languages in Web-pages, using L^AT_EX₂HTML*. TUGboat **19**(2), 195–203(1998).
3. Peter Signell: *One-document scientific publishing for print and Web/CD*. TUGboat **19**(2), 208–213(1998).
4. Michael D. Sofka: *TEX to HTML translation via tagged DVI files*. TUGboat **19**(2), 214–222(1998).
5. Lewis Carroll: **The Annotated Alice**. Alice's Adventures in Wonderland and Through the Looking-Glass, strana 122. Penguin Books 1970.
6. Zdeněk Wagner: *L^AT_EXová kuchařka/1*. Zpravodaj Československého sdružení uživatelů T_EXu, **6** (2), 96–108 (1996).
7. Zdeněk Wagner: *L^AT_EXová kuchařka/3*. Zpravodaj Československého sdružení uživatelů T_EXu, **7** (3), 140–167 (1997).
8. M. Goossens, F. Mittelbach, A. Samarin: **The L^AT_EX Companion**. Addison Wesley, Reading 1994. ISBN 0-201-54199-8.
9. Johannes Braams: *To reset or not to reset*. TUGboat **15**(4), 496–497(1994).
10. D. E. Knuth: **The T_EXbook**. Addison Wesley, Reading 1984. ISBN 0-201-13448-9.
11. P. Olšák: **T_EXbook naruby**. Konvoj, Brno 1997. ISBN 80-85615-64-9. Elektronická verze je k dispozici na <http://math.feld.cvut.cz/olsak/tbn/>.
12. Květoslav Minařík: **Pataňdžaliho jógasútra**. Canopus, Praha 1994. ISBN 80-85202-25-5.
13. W. Y. Evans-Wentz, K. Minařík: **Milaräpa, velký tibetský jógin**. Canopus, Praha 1996. ISBN 80-85202-28-X.

14. Eduard Petiška: **Châteaux et Légendes, Promenades au Royaume de Bohême et de Moravie**. Martin, Praha 1996. ISBN 80-85955-02-4.

Zdeněk Wagner
wagner@mbox.cesnet.cz

Kam za TUGboat-mi?

ŠTEFAN PORUBSKÝ

Časopis TUGboat bol založený v roku 1980 združením T_EX Users Group (<http://www.tug.org/>). Zájemcovia o toto periodikum sa môžu prihlásiť do tejto záujmovej organizácie na adrese

TeX Users Group
1466 NW Front Avenue, Suite 3141
Portland, OR 97209, USA

email: office@tug.org
fax: +1-503-223-3960
voice: +1-503-223-9994

a dostanú ho ako členský časopis. Členský príspevok je \$60 ročne.

ČSTUG je kolektívnym členom TUG (\$350 / rok) a z tohto titulu v súčasnosti dostáva 7 exemplárov. Časopisy sú rozmiestnené podľa nasledujúceho rozdeľovníka:

- 1 ks Matematicko-informatické oddelení MFF, Knihovna Václava Hlavatého (Karlín), Sokolovská 83, Praha 8 Karlín, tel. 02-2191 3292. Výpožičná doba knižnice a študovňa je otvorená v čase:

Po		9.00 – 18.00
Ut		9.00 – 18.00
St		9.00 – 18.00
Št		9.00 – 18.00
Pi		9.00 – 15.00

K dispozícii sú tieto ročníky alebo čísla:

- 1 ks knihovna MÚ AV ČR, Žitná 25, Praha 1. Výpožičná doba a študovňa je otvorená v čase:

Po		8.00 – 12.00	13.00 – 15.00
St		8.00 – 12.00	13.00 – 15.00
Pi		8.00 – 13.00	