

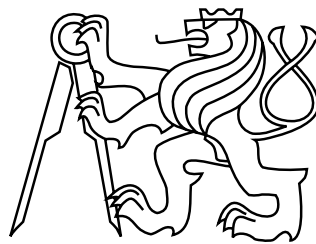
ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE  
FAKULTA STAVEBNÍ

## DIPLOMOVÁ PRÁCE

PRAHA 2009

Anna HORČIČKOVÁ

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE  
FAKULTA STAVEBNÍ  
OBOR GEODÉZIE A KARTOGRAFIE



DIPLOMOVÁ PRÁCE  
PREZENTAČNÍ MOŽNOSTI SYSTÉMU  $\text{\LaTeX}$

Vedoucí práce: Ing. Petr SOUKUP, Ph.D.  
Katedra mapování a kartografie

leden 2009

Anna HORČIČKOVÁ

ZDE VLOŽIT LIST ZADÁNÍ

Z důvodu správného číslování stránek

## ABSTRAKT

Práce se zabývá tématem *Prezentační možnosti systému  $\LaTeX$* . Obsahuje stručný popis systému  $\LaTeX$  a přehled vhodných editorů, které se při tvorbě textů v tomto systému používají. Dále obsahuje přehled šablon používaných v ČR pro psaní vysokoškolských studentských prací a popis vytváření šablony pro studentské práce na FSv ČVUT. Následující kapitola se zabývá nástroji systému  $\LaTeX$ , které lze použít k přípravě prezentací. Podrobněji se zmiňuje o třídě Beamer, ve které je vytvořena šablona určená pro prezentace prací na ČVUT. V závěrečné části práce jsou uvedeny možnosti automatického vytváření internetových stránek.

## KLÍČOVÁ SLOVA

Beamer,  $\LaTeX$ , LyX, šablona, TeXLive, Texmaker, tex4ht

## ABSTRACT

This thesis considers the *Presentation possibilities of a LaTeX system*. It contains a brief description of the  $\LaTeX$  and a summary of suitable editors that are used for writing texts in this system. Further to this, it also contains a summary of templates that are being used in the Czech Republic for writing students' thesis at FCE CTU in Prague. The following chapter of this thesis considers  $\LaTeX$  tools which can be used for preparing presentations. In more detail, it refers to a Beamer Class in which a template for presentation at CTU in Prague is created. In the final chapter there are stated possibilities of an automatic creation of web pages.

## KEYWORDS

Beamer Class,  $\LaTeX$ , LyX, template, TeXLive, Texmaker, tex4ht

## PROHLÁŠENÍ

Prohlašuji, že diplomovou práci na téma „Prezentační možnosti systému L<sup>A</sup>T<sub>E</sub>X“ jsem vypracovala samostatně. Použitou literaturu a podkladové materiály uvádím v seznamu zdrojů.

V Praze dne .....

.....

(podpis autora)

## PODĚKOVÁNÍ

Chtěla bych poděkovat vedoucímu práce Ing. Petru Soukupovi, Ph.D. za připomínky, inspiraci a pomoc při zpracování této práce. Dále bych ráda poděkovala Honzovi Jiránkovi za konzultace a psychickou podporu při řešení počátečních problémů se systémem  $\text{\LaTeX}$ .

# Seznam zkratek

|          |   |
|----------|---|
| ČVUT     | České vysoké učení technické                                |
| FCE      | Faculty of Civil Engineering                                |
| FEKT     | Fakulta elektrotechniky a komunikačních technologií         |
| FEL      | Fakulta elektrotechnická                                    |
| FSv      | Fakulta stavební  |
| CTU      | Czech Technical University                                  |
| VŠB – TU | Vysoká škola báňská – Technická univerzita Ostrava          |
| VUT      | Vysoké učení technické                                      |
| WYSIWYG  | What You See Is What You Get – Co vidíte, to dostanete      |
| WYSIWYM  | What You See Is What You Mean – Co vidíte, to máte na mysli |

# Obsah

|  |           |
|--|-----------|
| Úvod   | 1         |
| <b>1 <math>\LaTeX</math></b>                       | <b>3</b>  |
| 1.1 Instalace $\LaTeX$ u                           | 3         |
| 1.1.1 TeXLive 2007                                 | 3         |
| 1.1.2 Přidávání balíků                             | 4         |
| 1.1.3 Čeština                                      | 6         |
| 1.2 Textové editory vhodné pro $\LaTeX$            | 6         |
| 1.2.1 Texmaker                                     | 7         |
| 1.2.2 TeXnicCenter                                 | 9         |
| 1.2.3 WinShell                                     | 10        |
| 1.2.4 LyX  | 12        |
| <b>2 Šablony pro studentské práce</b>              | <b>15</b> |
| 2.1 Šablona 2.02, VUT FEKT v Brně                  | 15        |
| 2.1.1 Struktura a nastavení šablony                | 15        |
| 2.1.2 Funkční šablona                              | 16        |
| 2.1.3 Výhody a nevýhody šablony                    | 16        |
| 2.2 Šablony z ČVUT FEL v Praze                     | 17        |
| 2.2.1 Šablona Diplomka                             | 17        |
| 2.2.2 Šablona Diplomka2                            | 18        |
| 2.2.3 Další šablony                                | 18        |
| 2.3 Šablona VŠB – TU Ostrava                       | 19        |
| 2.3.1 Struktura šablony                            | 19        |
| 2.3.2 Zhodnocení třídy dokumentu diploma           | 21        |
| <b>3 Vytvoření šablony pro FSv ČVUT</b>            | <b>22</b> |
| 3.1 Obálka a titulní strana práce                  | 22        |
| 3.2 Prohlášení                                     | 24        |
| 3.3 Záhloví a zápatí                               | 24        |
| 3.4 Vytvoření příkazu pro vysázení měsíce obhajoby | 26        |
| <b>4 Prezentace pro publikum</b>                   | <b>27</b> |
| 4.1 Beamer   | 27        |
| 4.1.1 Instalace třídy a struktura adresářů         | 27        |
| 4.1.2 Čeština                                      | 29        |
| 4.1.3 Vzhled prezentace                            | 29        |



|          |   |           |
|----------|---|-----------|
| 4.1.4    | Předdefinované příkazy . . . . .  | 30        |
| 4.1.5    | Postup vytvoření prezentace . . . . .   | 31        |
| 4.2      | Další možnosti při tvorbě prezentací v L <sup>A</sup> T <sub>E</sub> Xu . . . . . | 33        |
| 4.2.1    | Seminar . . . . .   | 33        |
| 4.2.2    | Prosper, ha-prosper . . . . .   | 33        |
| 4.2.3    | Powerdot . . . . .  | 34        |
| <b>5</b> | <b>Šablona pro prezentace na FSv ČVUT</b>   | <b>35</b> |
| 5.1      | Doplnění šablony . . . . .  | 35        |
| 5.2      | Výběr tématu . . . . .  | 36        |
| <b>6</b> | <b>Webové stránky</b>   | <b>37</b> |
| 6.1      | Pokusy vytvoření internetových stránek . . . . .                                  | 37        |
| 6.1.1    | Editory . . . . .   | 37        |
| 6.1.2    | Příkazový řádek . . . . .   | 38        |
| 6.1.3    | ImageMagick . . . . .   | 39        |
| 6.1.4    | Soubor tex4ht.env . . . . .   | 39        |
| 6.2      | tex4ht a šablona pro studentské práce . . . . .                                   | 40        |
| 6.3      | Shrnutí . . . . .   | 41        |
|          | <b>Závěr</b>  | <b>43</b> |
|          | <b>Použité zdroje</b>   | <b>44</b> |
| <b>A</b> | <b>Změny v Šabloně 2.02</b>   | <b>I</b>  |
| A.1      | Obálka a titulní strana . . . . .   | I         |
| A.1.1    | Příkaz pro vytvoření obálky . . . . .   | I         |
| A.1.2    | Příkaz pro vytvoření titulní strany . . . . .                                     | IV        |
| A.2      | Řešení záhlaví . . . . .  | VII       |
| A.2.1    | Soubor <code>thesiscvut.sty</code> . . . . .                                      | VII       |
| A.2.2    | Soubor <code>sablona.tex</code> . . . . .   | VII       |
| A.2.3    | Ostatní soubory . . . . .   | VIII      |
| A.3      | Měsíc obhajoby . . . . .  | IX        |
| A.3.1    | Soubor <code>thesiscvut.sty</code> . . . . .                                      | IX        |
| A.3.2    | Soubor <code>sablona.tex</code> . . . . .   | IX        |
| <b>B</b> | <b>Prezentace</b>   | <b>X</b>  |
| B.1      | Použití některých příkazů . . . . .   | X         |
| B.1.1    | Příkaz <code>author</code> . . . . .  | X         |
| B.1.2    | Příkaz <code>institute</code> . . . . .   | X         |
| B.1.3    | Příkaz <code>AtBeginSubsection</code> . . . . .                                   | XI        |

|                                     |           |
|-------------------------------------|-----------|
| B.2 Jednoduchá prezentace . . . . . | XI        |
| <b>C Příložené CD</b>               | <b>XV</b> |

# Úvod

Téma této diplomové práce se odvíjí od snahy sjednotit a zlepšit vzhled studentských prací, které vznikají na FSv ČVUT v Praze, na oboru Geodézie a kartografie. Pokud vyjdeme z aktuální situace, kdy studenti nemají žádné směrnice, podle kterých by se mohli řídit při psaní prací, pak o jednotnosti a v některých případech i o kvalitě vzhledu práce nelze mluvit. Vzhled své práce si určuje každý student sám. Většina studentů se setká s psaním rozsáhlejšího textu poprvé v životě u vytváření své závěrečné práce. Proto vzniká rozmanitá škála prací, z nichž některé jsou opravdu kvalitní, ale jiné méně. Studenti se dopouštějí základních typografických chyb ve výběru písma, členění dokumentu, umístění obrázků a dalších. Díky těmto chybám některé práce ztrácí na kvalitě, i když jsou obsahově výborné.

Základním cílem této práce je vytvoření šablony, která by sloužila studentům jako vzor při psaní jejich prací. Vzhledem k tomu, že ČVUT je školou technickou, je vhodné pro tento záměr zvolit kvalitní sazečský program, který si dokáže poradit s nejrůznějšími vzorci, tabulkami a nákresey. Tato kritéria bezesporu splňuje typografický systém  $\text{\LaTeX}$ . Na technicky zaměřených školách je již řadu let k psaní závěrečných prací doporučován. Mezi jeho nevýhody lze zařadit to, že psaní textu se může zdát zpočátku poměrně složité. Uživatel si musí osvojit odlišné návyky od těch, které běžně používá u textových editorů (např. MS Word). Studenti Geodézie a kartografie mají možnost se se systémem  $\text{\LaTeX}$  seznámit v úlohách předmětu KAPR (Kartografická polygrafie a reprografie). Vypracováním několika úloh důvěru k  $\text{\LaTeX}$ u většina studentů bohužel nezíská a kvůli obavám z neznalosti systému si pro psaní závěrečné práce raději vyberou jiný program, který jim neumožní vytvořit vzhledově tak kvalitní práci. Také právě proto byl zvolen systém  $\text{\LaTeX}$  pro vytvoření šablony.

K vytvoření závěrečné práce patří obhajoba a s ní spojená prezentace práce před komisí. Systém  $\text{\LaTeX}$  umožňuje vytváření kvalitních prezentací. Proto součástí této práce je i kapitola, která se problematice prezentací věnuje. Opět je zde připravena šablona pro vytvoření jednoduché prezentace, kterou lze použít nejen při obhajobě závěrečné práce, ale po drobných úpravách i pro prezentaci práce naší školy.

Nedílnou součástí studentských prací je v dnešní době jejich zveřejnění na internetu.  $\text{\LaTeX}$  disponuje nástroji, které umožňují automatickou tvorbu internetových stránek.

Název práce *Prezentační možnosti systému  $\text{\LaTeX}$*  tedy zahrnuje řešení všech výše uvedených problémů.

Práci jsem rozdělila do šesti kapitol. První kapitola se zabývá systémem  $\text{\LaTeX}$  jako takovým. Popisuje jeho instalaci, rozšiřování o různé balíky, přehled a popis dostupných editorů, které lze pro vytváření textů v  $\text{\LaTeX}$ u použít. Druhá kapitola

obsahuje popis několika šablon, které vznikly na vysokých školách v České republice. Třetí kapitola se zabývá vytvořením šablony pro psaní studentských prací na FSv ČVUT v Praze. Čtvrtá kapitola seznamuje čtenáře s možnostmi vytváření prezentací pomocí systému  $\text{\LaTeX}$ . Lze v ní najít stručný postup, jak jednoduše prezentaci vytvořit. Pátá kapitola se zabývá přípravou šablony pro vytváření prezentací na FSv ČVUT v Praze. Poslední, šestá kapitola, popisuje nástroje pro automatickou konverzi textu z  $\text{\LaTeX}$ u do formátů použitelných při zveřejnění práce na internetu. Neopodstatitelnou částí této práce je přiložené CD, na kterém jsou k dispozici všechny potřebné soubory.

# 1 $\LaTeX$

$\LaTeX$  je typografický systém pro sázení textu. Byl vytvořen jako nadstavba k systému  $\TeX$ . Systém  $\TeX$  vytvořil Donald E. Knuth v 70. letech 20. století.  $\TeX$  se řadí k tzv. WISIWYM systémům pro psaní textu.  $\TeX$  je určen pro sázení technických textů, které obsahují vzorce a tabulky. Je v něm definováno přibližně 300 základních příkazů. Obsahuje nástroje k tvorbě nových příkazů. Díky nim vznikají nové nadstavby jako např.  $\LaTeX$ .

$\LaTeX$  je pro běžného uživatele srozumitelnější. Vznikl v 80. letech 20. století. Vynul ho Leslie Lamport. Umožňuje jednodušší a přirozenější zápis sázeného textu. Příprava textu v  $\LaTeX$ u se podobá programování. V libovolném textovém editoru se vytvoří zdrojový soubor. Tento soubor obsahuje nejen vlastní text, ale i příkazy, které určují, jak bude výsledný text vysázen. Příkazy, které ovlivňují vlastnosti celého dokumentu, se zadávají na začátku souboru v hlavičce (preambuli) souboru. Takovými příkazy se přidávají balíky (viz. 1.1.2), volí se způsob sázení češtiny (viz. 1.1.3), určuje se třída dokumentu, velikost papíru, ... Vlastní text (včetně příkazů určujících vzhled částí textu – nadpisy, zarovnání, ...) se píše do těla souboru, které je ohraničeno začátkem (`\begin{document}`) a koncem dokumentu (`\end{document}`). Zdrojový soubor je přeložen překladačem  $\TeX$  s nadstavbou  $\LaTeX$  do výsledného netextového souboru (`*.pdf`, `*.dvi`), který lze zobrazit pomocí nějakého prohlížeče (**Acrobat Reader**, **GSView**). Dalším výstupním souborem je soubor `*.log`. Tento soubor obsahuje zprávy o průběhu překladače včetně chybových a varovných hlášení.

## 1.1 Instalace $\LaTeX$ u

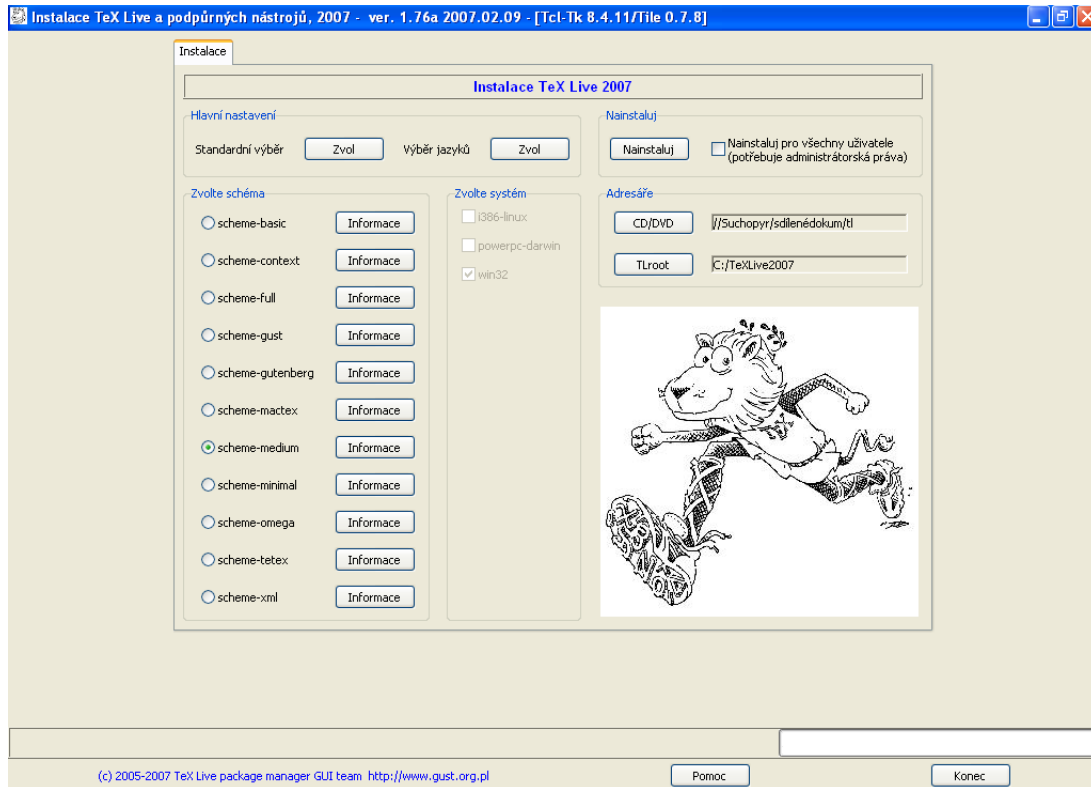
Instalace  $\LaTeX$ u se liší podle vybrané distribuce. Distribuce je souhrn souborů, které uživatel potřebuje pro správnou instalaci a funkci systému. Existují různé typy distribucí. Některé jsou vhodné pouze pro unixové/linuxové systémy (`teTeX`), jiné pro systémy Windows (`MiKTeX`), dále pro operační systémy Linux, Windows, MacOS (`TeXLive`).

### 1.1.1 `TeXLive 2007`

`TeXLive 2007` je tedy distribuce vhodná pro všechny hlavní operační systémy. Distribuuje se na CD/DVD a jeho výhodou je, že lze vše spustit z tohoto nosiče. Dále je možno stáhnout soubor `t12007.zip` s obsahem distribučního CD/DVD z internetu a `TeXLive` nainstalovat do počítače. Součástí instalace je kromě  $\LaTeX$ u i adresář s podpůrnými programy, kde je uložen editor pro  $\LaTeX$ , prohlížeč výstupních souborů ve formátu `*.dvi` a další programy. Postup instalace `TeXLive 2007`

do všech operačních systémů je popsán v dokumentaci, která je uložena v adresáři: `tl/texmf-doc/doc/czechoslovak/texlive-cz`.

Pro instalaci ve Windows se spustí soubor `tlpmgui.exe`. Otevře se okno s názvem Instalace TeXLive a podpůrných nástrojů (viz Obr. 1.1). V tomto okně uživatel



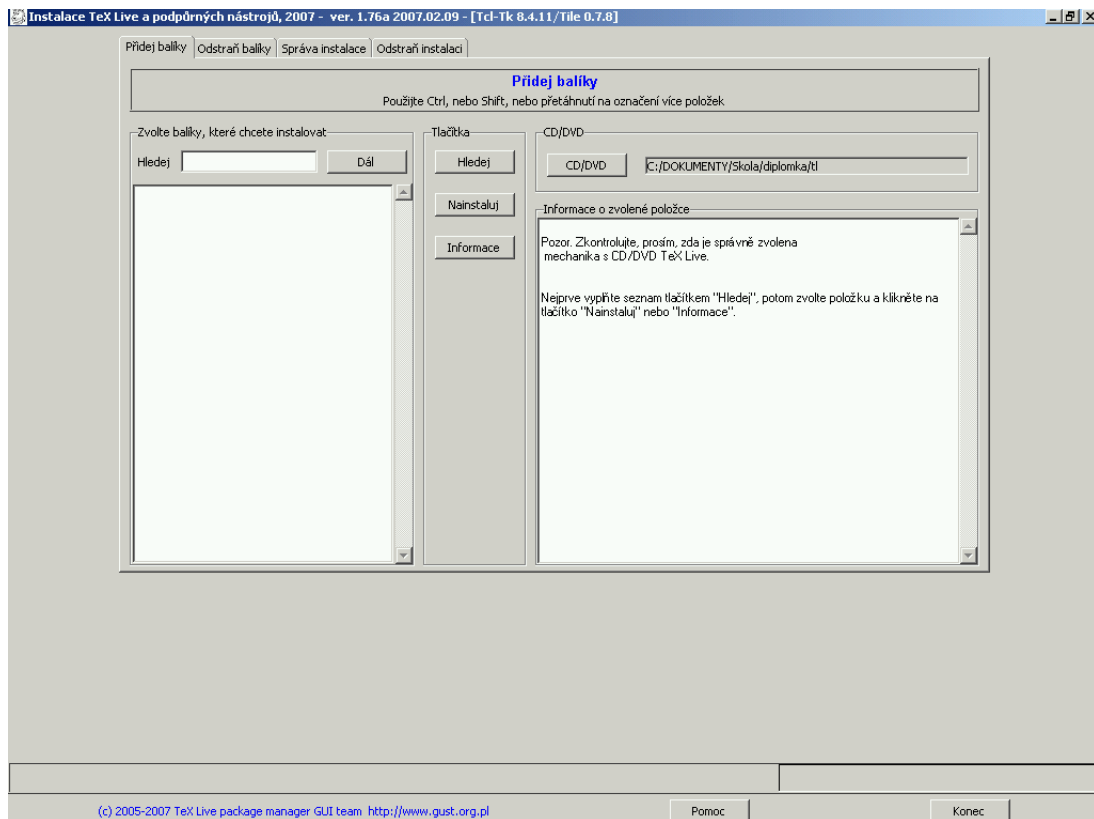
Obr. 1.1: Průvodce instalací TeXLive 2007

nastaví schéma, které chce nainstalovat. Jednotlivá schémata obsahují různé typy instalovaných programů. Obsah schémat lze zjistit kliknutím na tlačítko **Informace** v příslušném řádku. Další možnosti nastavení jsou v sekci **Hlavní nastavení**. Zde uživatel vybírá jazyky, příp. některé další součásti L<sup>A</sup>T<sub>E</sub>Xu jako např. dokumentaci v různých jazycích. Po nastavení se pouze stiskne tlačítko **Nainstaluj**. Průvodce instaluje i podpůrné programy, na jejichž instalaci se dotazuje, proto je potřeba občas stisknout tlačítko, aby instalace pokračovala. Po dokončení instalace se nikde v počítači neobjeví ikona L<sup>A</sup>T<sub>E</sub>Xu, kterou by se program spouštěl. Program se volá příkazem `latex dokument.tex` v příkazovém řádku, kde `dokument.tex` je zdrojový text dokumentu.

### 1.1.2 Přidávání balíků

Při práci s L<sup>A</sup>T<sub>E</sub>Xem uživatel potřebuje používat různé balíky, které vždy nemusí být součástí nainstalovaného výběru a je tedy nutné je přidat. V TeXLive 2007 je pro

tento případ připraven průvodce. Jedná se o stejného průvodce jako při instalaci, ale vzhled výchozího okna je jiný (viz Obr.1.2). Toto okno obsahuje čtyři záložky.



Obr. 1.2: Prostředí pro přidávání a odebírání balíků pro TeXLive 2007

V záložce **Přidej balíky** v sekci **CD/DVD** je třeba zvolit adresář, ve kterém je umístěn obsah instalačního CD/DVD. Nejedná se tedy o adresář **TeXLive2007**, který vznikne nainstalováním, ale o adresář **tl**. Po nastavení správného adresáře se stiskne tlačítko **Hledej**. Tím se vyhledají všechny nenainstalované části systému a zobrazí se v levém okně průvodce. Zde uživatel vybere balíky, které chce doinstalovat a klikne na tlačítko **Nainstaluj**. Po úspěšně dokončené instalaci balíků je nutné v záložce **Správa instalace** stisknout tlačítko **Obnov** v sekci **Obnov databázi ls-R**. Pak už by měl být přidán balík funkční.

Může se stát, že balík, který potřebuje uživatel přidat, není součástí instalace. Pak je třeba ho najít na internetu (nebo získat nějak jinak) a nakopírovat do adresáře **texmf-local**. Tento adresář obsahuje několik podadresářů. Pro lepší orientaci je vhodné dodržovat předdefinovanou strukturu a ukládat tedy balíky do adresáře **texmf-local/tex/latex**, dokumentaci do **texmf-local/doc**, fonty do **texmf-local/fonts** atd.

Seznam a popis dosud vytvořených balíčků včetně dokumentace je uveden na stránkách CTAN: View package information<sup>1</sup>, ze kterých lze balíky také stahovat.

### 1.1.3 Čeština

Aby mohl být text psaný v L<sup>A</sup>T<sub>E</sub>Xu „dokonalý“ i po stránce češtiny, je nutné zajistit nejen sázení českých znaků (jako je např. ě, š, ů, ...) a českých uvozovek, ale také docílit toho, aby se správně dělila slova a česky sázely předdefinované názvy např. kapitol, sekcí a nebo měsíců u data generovaného pomocí příkazu `\today`. Tyto požadavky lze v L<sup>A</sup>T<sub>E</sub>Xu obecně zajistit dvěma způsoby.

První z nabízených možností je použití balíčku `babel` s parametrem `czech`. Tento způsob se prozatím jeví jako hlavní, který se vyvíjí a bude do budoucna využíván. Používá k překladu zdrojového textu originálního programu `latex.exe` (příp. `pdflatex.exe`, ...). Tím je zajištěna univerzálnost použití na různých instalacích L<sup>A</sup>T<sub>E</sub>Xu a možnost vývoje balíčků s dalšími jazyky. Vložení balíku `babel` se zajistí příkazem:

```
\usepackage[babel]{czech}.
```

V dnešní době je `babel` funkční, ale našla jsem na internetu stížnosti uživatelů ohledně slovní zásoby a dělení slov.

Druhou možností, jak sázet české znaky, je použití balíčku `czech.sty` a překládat zdrojový text pomocí programu `cslatex.exe` (příp. `pdfcslatex.exe`, ...), což je tzv. počeštěný překladač. Tato možnost je nyní funkční, ale není univerzální. Pokud na počítači není instalován program `cslatex.exe`, tak se zdrojový soubor nepřeloží správně. Další problémy se vyskytují při použití některých balíčků. Do zdrojového souboru se vloží příkazem v hlavičce:

```
\usepackage{czech}.
```

Více informací o sázení českých textů je k dispozici např. na stránkách: Jak sázet česky<sup>2</sup> a Něco málo o LaTeXu<sup>3</sup>.

## 1.2 Textové editory vhodné pro L<sup>A</sup>T<sub>E</sub>X

Z podstaty L<sup>A</sup>T<sub>E</sub>Xu vyplývá nutnost použití nějakého textového editoru. Existuje jich celá řada. V dalším textu jsou popsány některé z nich, které jsem měla možnost vyzkoušet. Popisují je z pohledu začátečníka. Uvedené editory jsem vyzkoušela pouze na OS Windows. Všechny tyto programy jsou volně k dispozici na internetu.

<sup>1</sup><http://tug.ctan.org/pkg/>

<sup>2</sup><http://www.fit.vutbr.cz/~martinek/latex/czech.html>

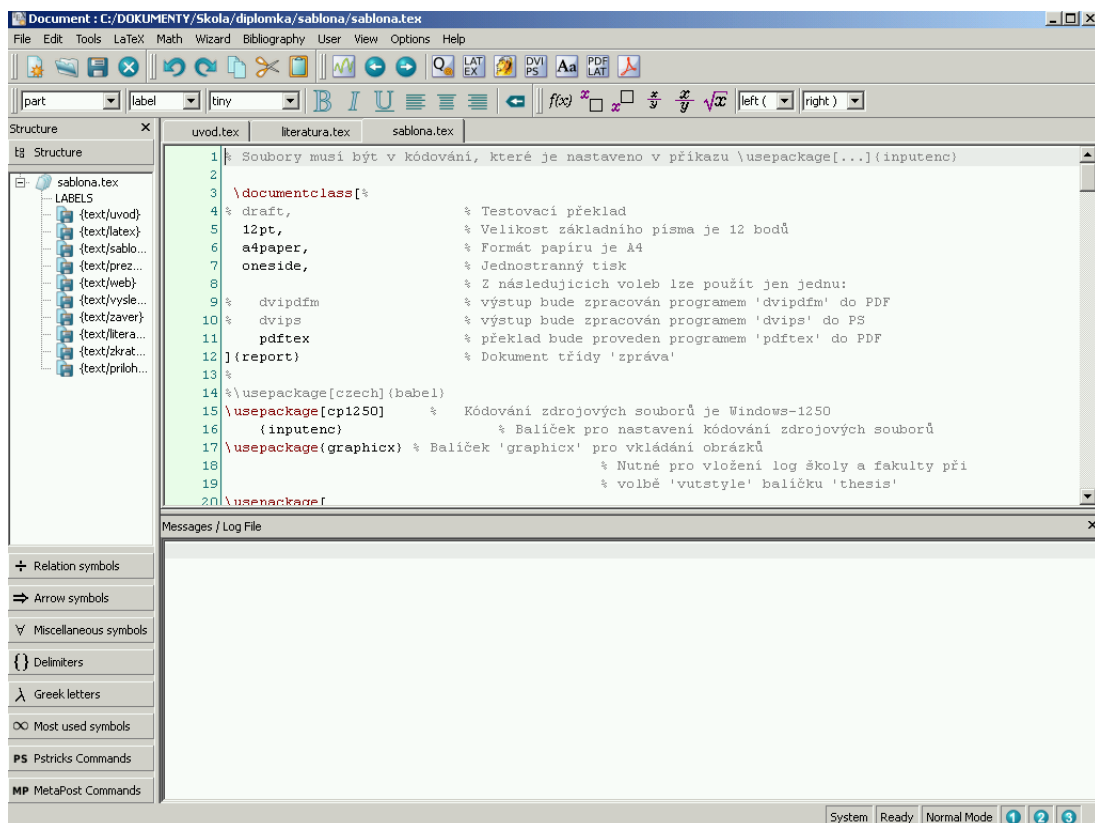
<sup>3</sup><http://fyzika.feec.vutbr.cz/~petrsad/LaTeX/cestina.htm>



## 1.2.1 Texmaker

Texmaker verze 1.8 je příjemný editor, který podporuje psaní v L<sup>A</sup>T<sub>E</sub>Xu. Na internetu je volně ke stažení na stránkách: Texmaker: Free LaTeX Editor.<sup>4</sup> Velikost instalačního balíčku je přibližně 6 MB. Po nainstalování zabere na disku místo 19 MB. Můžeme ho nainstalovat v operačních systémech Windows, Linux a Mac.

Základní prostředí Texmakeru (viz Obr. 1.3) je velmi přehledné, může obsahovat

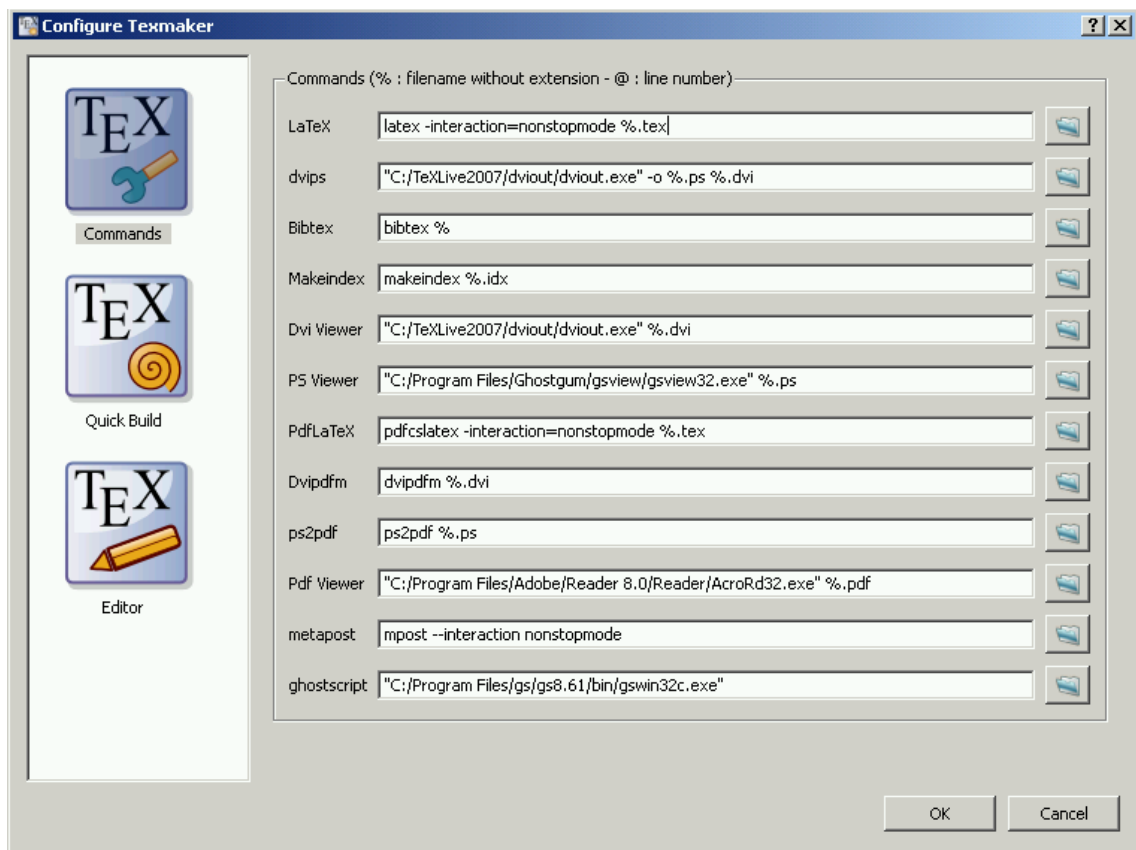


maximálně tři okna. Editační okno je nejvýraznější, slouží k prohlížení a editaci souborů. Lze v něm otevřít několik souborů najednou a přepínat mezi nimi pomocí záložek. Dalším oknem je okno **Messages/Log File**, které slouží ke zobrazování zpráv ohledně chodu kompilace. Při použití volby **Quick Build** se v tomto okně zobrazí text ze souboru \*.log. Posledním oknem je okno s názvem **Structure**, v němž se zobrazuje struktura právě otevřeného dokumentu.

Texmaker usnadní psaní v L<sup>A</sup>T<sub>E</sub>Xu připravenými příkazy. Ve formě tlačítek se v panelech nástrojů zobrazují příkazy pro zarovnávání textu, psaní matematických výrazů, vytvoření nového řádku, překladu textu různými programy (latex.exe,

<sup>4</sup><http://www.xmlmath.net/texmaker/download.html>

pdf<sub>l</sub>atex.exe), otevření prohlížeče výsledného souboru. Lze zde najít i konfigurovatelné tlačítko **Quick Build**, jehož funkce je popsána v dalším textu. Panel nástrojů dále obsahuje standardní tlačítka **Nový soubor**, **Otevřít soubor**, **Uložit soubor**, **Zavřít soubor**, **Zpět**, **Vpřed**, **Kopírovat**, **Vymout**, **Vložit**. Další nástroje jsou ukryty v hlavním menu pod záložkami **Tools**, **LaTeX**, **Math**, **Wizard** a **Bibliography**. Pod záložkou **User** se ukrývají možnosti nastavení různých klávesových zkratk. Velmi zajímavá a k správné funkčnosti nepostradatelná je záložka **Options**, pod kterou se skrývá volba **Configure Texmaker** (viz Obr. 1.4). Nastavení



Obr. 1.4: Texmaker – Nastavení

Texmakeru spočívá ve třech krocích. Ve volbě **Commands** je třeba nastavit příkazy, kterými se spustí překlad zdrojového souboru při různých druzích překladu. Dále se nastaví cesta k programům, které spustí prohlížeč výsledného souboru. Ve volbě **Quick Build** se nastaví souhrn příkazů, které se mají spustit po stisknutí tlačítka Quick Build (např.: PdfLaTeX + View PDF – přeložení zdrojového souboru pomocí pdf<sub>l</sub>atex, spuštění Acrobat Readeru a otevření výsledného souboru). Ve volbě **Editor** se nastaví vzhled Texmakeru. Řeší se zde velikost a font použitého písma, barva zvýraznění příkazů, klávesové zkratky pro některé zvláštní znaky či pro spuštění různých funkcí Texmakeru. Dále je zde možnost nastavení slovníku pro

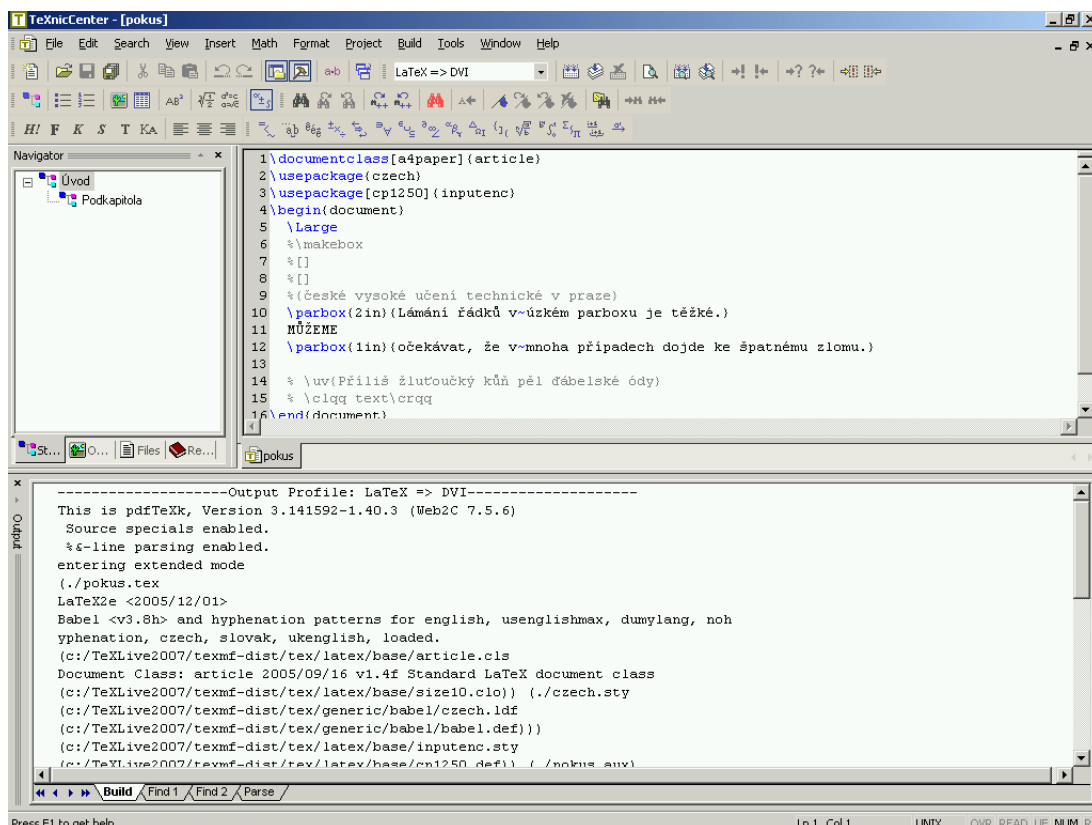
kontrolu pravopisu. Slovník je třeba nejprve stáhnout z internetu, uložit na disk, nastavit správnou cestu v Texmakeru a v hlavním menu v položce Edit zvolit Spelling Check. Kontrola pravopisu probíhá od pozice kurzoru.

S Texmakerem se mi pracuje velmi dobře. Zvýraznění párové závorky je u Texmakeru novinkou, je zavedeno pouze ve verzi 1.8. Tento program lze nainstalovat pouze anglicky.

## 1.2.2 TeXnicCenter

TeXnicCenter je volně šiřitelný program vytvořený pro psaní textů v L<sup>A</sup>T<sub>E</sub>Xu. Na internetu je volně ke stažení na stránkách: Download TeXnicCenter.<sup>5</sup> Velikost instalačního souboru je 4,42 MB. Nainstalovaný TeXnicCenter zabere na disku přibližně 12 MB. Je vytvořen pro operační systém Windows.

Základní prostředí programu TeXnicCenter (viz Obr. 1.5) je poměrně přehledné.



Obr. 1.5: TeXnicCenter – Základní prostředí

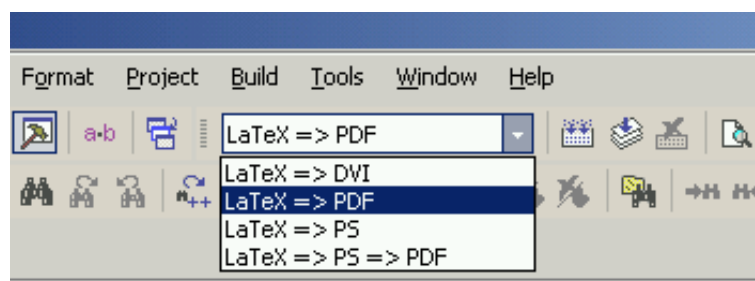
Na první pohled uživatele jistě zaujme velké množství ikon v panelech nástrojů. Nacházejí se zde tlačítka pro standardní příkazy typu **Otevřít soubor**, pro změnu

<sup>5</sup><http://www.toolscenter.org/resources/required-components/29>

velikosti písma, pro psaní matematických vzorců, vytváření číslovaných (nečíslovaných) seznamů, zarovnávání textu, hledání slov (posloupnosti znaků) a další užitečné funkce. Samozřejmě lze panely nástrojů upravit tak, aby obsahovaly pouze ikony, které si uživatel přeje. Všechny nástroje jsou skryty pod nabídkami hlavního menu.

Základní prostředí obsahuje tři okna. Editační okno, ve kterém je možné otevřít několik souborů a přepínat mezi nimi pomocí záložek. Lepší orientaci v textu zajišťuje barevné zvýraznění příkazů a hledání párové závorky. Další okno, které se jmenuje **Navigator**, obsahuje několik záložek. Lze zde najít strukturu textu, připojené soubory, objekty (obrázky, tabulky, ...). Ve třetím okně **Output** se zobrazuje hlášení o průběhu překladu.

Překlad zdrojového textu lze spustit tlačítkem v panelech nástrojů, klávesou F7, nebo vybrat z nabídky pod volbou **Build**. Před spuštěním překladu je nutné nastavit druh výsledného souboru v rolovacím menu v panelech nástrojů (viz Obr. 1.6).



Obr. 1.6: TeXnicCenter – Nastavení formátu výstupního souboru

Instalace programu TeXnicCenter je výborně krok po kroku popsána na stránkách: Instalace TeXnicCenter.<sup>6</sup> Při prvním spuštění programu se zobrazí průvodce nastavením, který umožní přehledně nastavit TeXnicCenter. Nastavení se týká hlavně cest k programům, které spouští překlad zdrojového textu. Na první pohled pro začátečníka poměrně složitý program si jistě po základním prohlédnutí oblíbí každý uživatel. TeXnicCenter lze nainstalovat v angličtině a němčině.

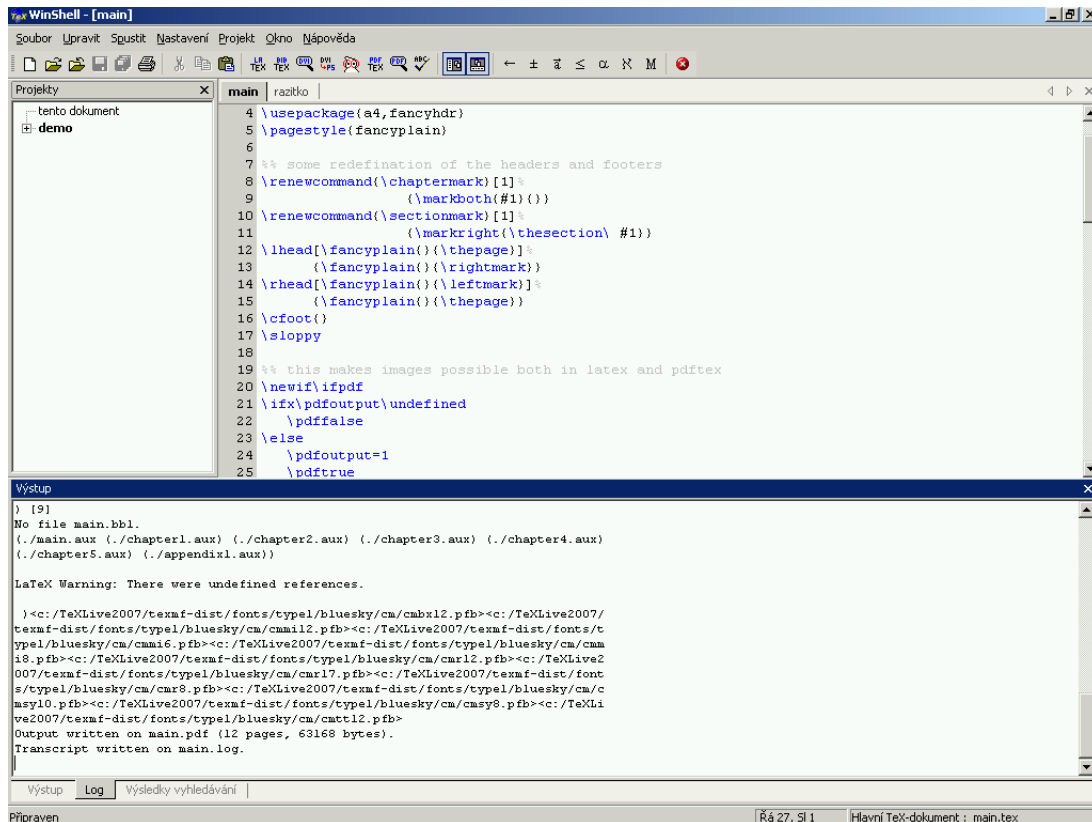
### 1.2.3 WinShell

Editor WinShell je určený pro vytváření a úpravu textu v systému T<sub>E</sub>X a L<sup>A</sup>T<sub>E</sub>X pod Windows. Je volně dostupný na internetu: WinShell – Download<sup>7</sup>, nebo ho lze nainstalovat z distribuce TeXLive 2007, kde je uložen v adresáři: `t1/support/winshell`. Instalační balíček je velký 4,1 MB. Nainstalovaný WinShell zabírá místo 4,3 MB.

Základní prostředí WinShellu (viz Obr.1.7) je srovnatelné s předchozími dvěma programy. Opět obsahuje tři okna. Editační okno, ve kterém lze otevřít více souborů

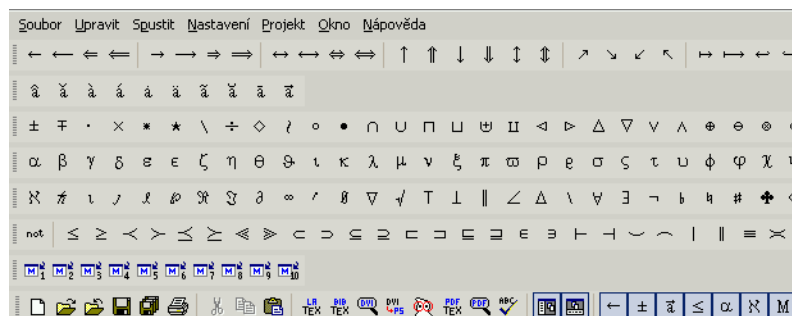
<sup>6</sup><http://latex.feec.vutbr.cz/cz/latex/lokalni-instalace/instalace-texniccentra/>

<sup>7</sup><http://www.winshell.org/modules/>



Obr. 1.7: Winshell – Základní prostředí

a přepínat mezi nimi pomocí záložek. Orientaci v textu usnadňuje barevné zvýraznění příkazů, párových závorek, skupin, matematického prostředí a komentářů. Barvy si může uživatel definovat dle svého uvážení. Okno **Projekty**, kde lze najít umístění otevřeného aktivního souboru. Okno **Výstup** se třemi záložkami **Výstup**, **Log** a **Výsledky vyhledávání**. Obsah těchto záložek je zřejmý z jejich názvu.

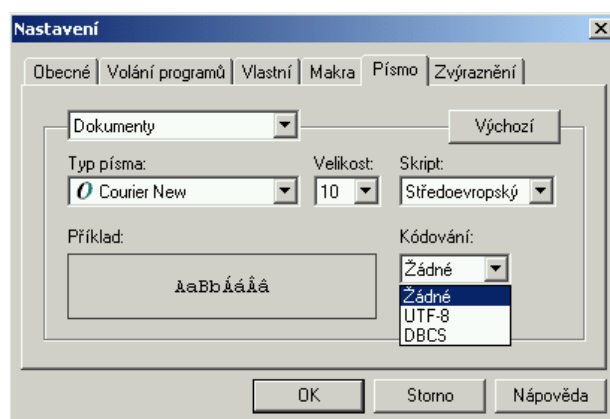


Obr. 1.8: Winshell – Výřez z panelů nástrojů

Panely nástrojů (viz Obr. 1.8) jsou zpracovány velmi přehledně a jednoduše. V hlavním okně WinShellu je pouze jeden řádek s ikonami. Levá část panelu je vyhrazena pro standardní úkony typu **Otevřít soubor**, **Uložit soubor**, ... Pro-

střední část obsahuje tlačítka pro překlad zdrojového textu a pro spuštění programů pro prohlížení výsledných dokumentů. V pravé části lze najít všechny ostatní podstatné nástroje. Obsahuje ikony pro otvírání a zavírání oken **Projekty** a **Výstup** a vždy po jedné ikoně pro šipky, binární operátory, akcenty, relace, řeckou abecedu, matematické vzorce. Kliknutím na jednu z těchto ikon se otevře další panel s výběrem příslušných nástrojů.

WinShell má pěkně zpracované možnosti v nastavení (viz Obr. 1.9). Nabídka **Nastavení** obsahuje 6 záložek. Velmi podstatnou záložkou je jistě **Volání programů**, kde se nastaví cesta k programům, které spouští překlad zdrojového souboru. Další důležitou záložkou je **Písmo**, ve které lze nastavit nejen font a stupeň písma v editoru, ale hlavně skript a kódování. To je velmi podstatné pro správné zobrazování českých znaků.



Obr. 1.9: WinShell – Možnosti nastavení

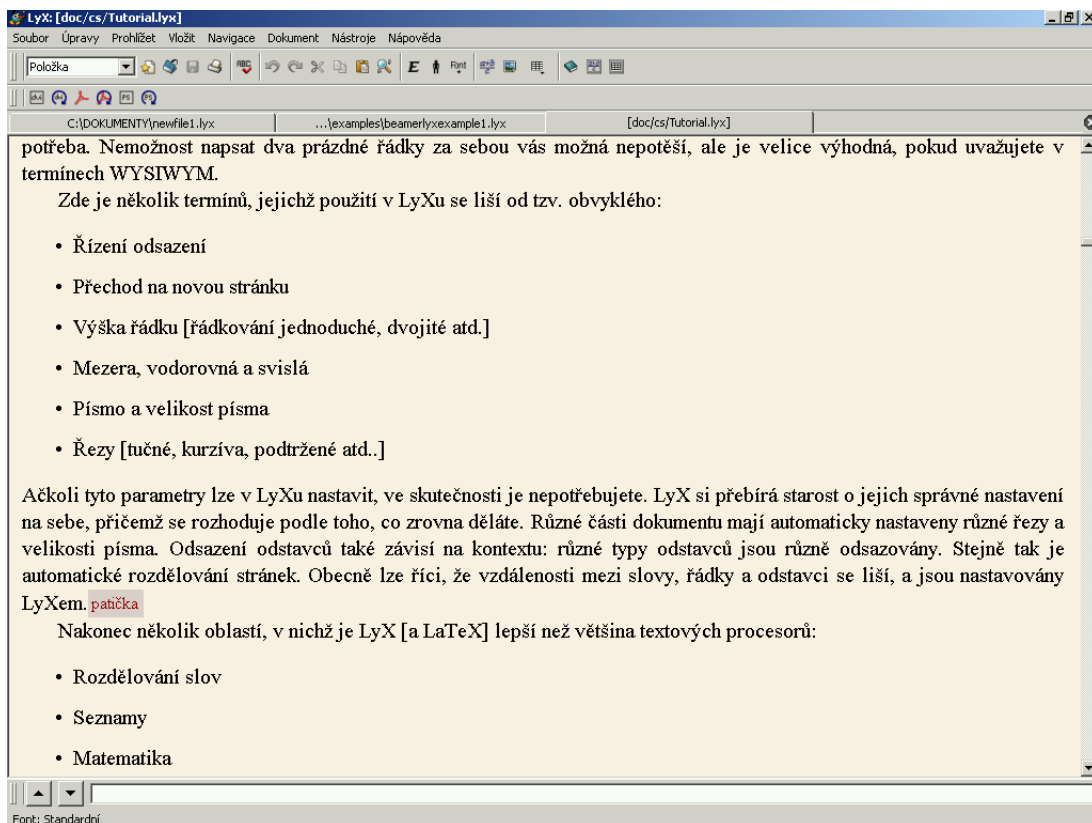
WinShell je možné nainstalovat v několika jazycích, mezi nimiž nechybí čeština. Doporučila bych ho lidem, kteří už tuší, jak se s  $\LaTeX$ em pracuje. Obsahuje nástroje k tvorbě vlastních maker. Více informací o tomto programu obsahují stránky: WinShell –  $\LaTeX$ User Front End<sup>8</sup>.

### 1.2.4 LyX

LyX je velmi zajímavý editor. Spojuje výhody  $\LaTeX$ u a klasického textového procesoru (např. Microsoft Word). Na první pohled se jeví jako WYSIWYG editor, ale není tomu tak, řadí se mezi tzv. WYSIWYM editory. Proto je pro uživatele, který je zvyklý na Microsoft Word a chce sázet dokonalejší texty, ideální volbou. Je volně ke stažení na internetu: LyX—download<sup>9</sup>. Velikost instalačního souboru je 19,6 MB. Nainstalovaný LyX zabere na disku 80,5 MB.

<sup>8</sup><http://www.winshell.org/>

<sup>9</sup><http://www.lyx.org/Download>



Obr. 1.10: LyX – Základní prostředí

Základní prostředí LyXu (viz Obr. 1.10) je přehledné, obsahuje jedno okno, ve kterém se otevírají soubory. Pokud je otevřeno více dokumentů, je možné mezi nimi jednoduše přepínat. Nabídka hlavního menu je česky, což je jistě vítáno. Hlavní menu obsahuje vše, co si běžný uživatel může přát. Panel nástrojů je v základním nastavení velmi strohý a tím pádem i přehledný. Obsahuje pouze několik ikon, které lze použít k překladu dokumentu pomocí L<sup>A</sup>T<sub>E</sub>Xu, standardní ikony pro práci se soubory (**Otevřít**, **Uložit**, **Vytvořit nový**, ...) a několik ikon, které umožní přidání panelu nástrojů pro editaci a vytváření matematických vzorců a tabulek. Samozřejmě lze v LyXu zapnout další panely nástrojů přes hlavní menu, volba Prohlížeč a dále volba Panely nástrojů.

Ovládání LyXu je do jisté míry intuitivní. Psaní textu se od Wordu liší např. tím, že pro vytvoření větší mezery mezi slovy nebo odstavci nelze použít několikanásobné stisknutí mezerníku nebo enteru. Mezery jsou dané, určuje je L<sup>A</sup>T<sub>E</sub>X. Pokud by je chtěl uživatel měnit, tak to je samozřejmě možné, ale je třeba, aby měl s L<sup>A</sup>T<sub>E</sub>Xem nějaké zkušenosti, aby ovládal základní příkazy.

Dokumentace k LyXu je zastoupena velmi pěkně zpracovanou nápovědou. Nápověda má několik částí. Většina je v angličtině. Výborným pomocníkem je část nazvaná Průvodce LyXu, která je přeložena do češtiny. V této části se uživatel

seznámí se základními informacemi, které jsou pro psaní v LyXu podstatné. Obsahuje i několik příkladů, na kterých se lze naučit a hlavně pochopit, co můžeme od LyXu očekávat a jak v něm vytvořit dokument podle svých představ. Dále tato část obsahuje odkazy do anglických částí nápovědy, které doplňují základní informace uvedené v Průvodci.



## 2 Šablony pro studentské práce

Vhodným pomocníkem při psaní studentských prací jsou šablony. Pomohou studentovi rozvrhnout práci podle daných pravidel a student se může lépe soustředit na vlastní obsah práce.

Pravidla psaní studentských prací jsou na jednotlivých školách definována různě. Některé školy mají přesné požadavky na to, jak by měla výsledná práce vypadat. Na webových stránkách Fakulty stavební ČVUT v Praze se mi nepodařilo najít dokument, který by definoval jednotná pravidla pro psaní závěrečných prací. Ke struktuře diplomové práce se vyjadřuje *Směrnice děkana pro realizaci studijních programů a státní závěrečné zkoušky na Fakultě stavební ČVUT v Praze*<sup>1</sup> v Článku 5, odstavec 8.

Na internetu se mi podařilo vyhledat několik šablon vhodných pro tvorbu studentských prací v  $\text{\LaTeX}$ u z různých škol.

### 2.1 Šablona 2.02, VUT FEKT v Brně

Tato šablona byla vytvořena Petrem Syslem a Pavlem Rajmicem nejen pro potřeby studentů na VUT FEKT (Vysoké učení technické – Fakulta elektrotechniky a komunikačních technologií) v Brně.

Jedná se o šablonu pro psaní semestrálních, bakalářských a diplomových prací v  $\text{\LaTeX}$ u. Tato šablona je velmi univerzální, má srozumitelnou dokumentaci a na internetu lze najít mnoho návodů k řešení problémů, které mohou při práci nastat. Šablona 2.02 je volně k dispozici na stránkách: Nová šablona (2.02).<sup>2</sup>

#### 2.1.1 Struktura a nastavení šablony

Šablona 2.02 se skládá ze dvou hlavních částí a to z balíčku `thesis.sty`, ve kterém jsou uloženy definice příkazů, vzhled úvodních stran práce, ... a z hlavního souboru `sablona.tex`. Dále šablona obsahuje adresář `text`, ve kterém se nacházejí jednotlivé části práce (Úvod, Řešení studentské práce, Výsledky studentské práce, Závěr, ...). Pokud studentovi vyhovuje styl, kterým je práce vysázena, zasahuje pouze do souboru `sablona.tex` a do jednotlivých částí práce v adresáři `text`.

Šablona umožňuje nastavení některých parametrů. Nastavitelné parametry shrnuji v následujícím výčtu.

- **Volba kompilátoru**

<sup>1</sup><http://www.fsv.cvut.cz/legislat/smedek07.php>

<sup>2</sup><http://latex.feec.vutbr.cz/cz/latex/download/nova-sablona/>

- `dvipdfm` – překlad bude proveden '`dvipdfm`' do PDF
- `dvips` – překlad bude proveden '`dvips`' do PS
- `pdftex` – překlad bude proveden programem '`pdftex`' do PDF

- **Styl práce**

- `english` – vysází úvodní stránky v angličtině
- `utstyle` – vysází úvodní stránky podle směrnice rektora VUT (obsahují logo školy a fakulty, jsou nastaveny okraje, ...)
- žádný – vysází úvodní stránky v češtině s obecným vzhledem (bez loga školy)

Styl `english` lze vybrat pouze při vypnuté volbě `utstyle`.

- **Druh práce**

- `diploma` – Diplomová práce
- `bachelor` – Bakalářská práce
- `semestral` – Semestrální práce

### 2.1.2 Funkční šablona

Pro správnou funkci šablony je nutné si podrobně prostudovat příložený návod. Obsahuje informace o tom, jak správně nastavit hlavní dokument a připojit balíček `thesis.sty`. Velmi podstatnou částí návodu je kapitola nazvaná *Závislost na ostatních balíčcích*. Obsahuje výčet balíčků nutných pro správnou funkci šablony. Pokud některý z uvedených balíčků není nainstalován, je potřeba ho najít a nainstalovat. V této kapitole je také zmíněno řešení češtiny přes balíček `czech.sty`. Je tedy nutné dokument překládat pomocí příkazu `pdfcslatex.exe` (`cslatex.exe`). Při hledání chybějících balíčků je vhodné podívat se nejprve do instalace  $\text{\LaTeX}$ u. Některé balíčky instalace obsahuje, ale nebyly nainstalovány vzhledem ke zvolenému typu instalace a lze je jednoduše doinstalovat (viz. kapitola 1.1). Další možností při hledání vhodných balíčků je samozřejmě internet. V obou případech je nutné kontrolovat verzi balíčků.

### 2.1.3 Výhody a nevýhody šablony

Mezi výhody této šablony bych určitě zařadila její univerzálnost. Šablona je použitelná pro jakýkoliv text. Volbami na začátku šablony uživatel pouze nastaví druh práce, styl práce a program, kterým se bude vytvářet výsledný dokument. Dále vyplní název práce, jméno autora, jméno vedoucího práce, obor studia, katedru, rok

a místo obhajoby. Po nastavení a vyplnění všech potřebných údajů se automaticky vysází titulní strana a obálka práce, další text je přizpůsoben zvolenému stylu. Šablona má velmi dobře zpracovanou dokumentaci a dokonce i své vlastní diskuzní fórum na stránkách: Diskuze<sup>3</sup>, kde lze najít mnoho zajímavých postřehů uživatelů.

U Šablony 2.02 postrádám obsah záhlaví a zápatí. V zápatí je pouze číslo stránky, záhlaví je prázdné. Podle mého názoru vhodně vyplněné záhlaví a zápatí oživuje práci a usnadňuje čtenáři orientaci v textu.

## 2.2 Šablony z ČVUT FEL v Praze

Do této kapitoly bych zahrнула několik šablon, které vznikly na Fakultě elektrotechnické na ČVUT v Praze. Jsou k dispozici na stránkách: Výuka – LaTeX<sup>4</sup>, kde je i několik zajímavých informací o L<sup>A</sup>T<sub>E</sub>Xu.

### 2.2.1 Šablona Diplomka

Šablonu Diplomka vytvořil Jiří Roubal. Již podle názvu je patrné, že je určena k psaní diplomových prací. Struktura šablony není úplně jednoduchá, ale po prvním prozkoumání uživatel zjistí, že má svou logiku. Adresář Diplomka obsahuje několik podadresářů a souborů:

- Adresáře:

Obsahem většiny adresářů jsou zdrojové soubory pro jednotlivé kapitoly. V každém adresáři je speciální složka pro obrázky vkládané do dané kapitoly. Na první pohled se může takové členění zdát zbytečné, ale při psaní delšího dokumentu jistě najde své uplatnění.

Adresář `Styles` obsahuje některé styly, které jsou k šabloně připojeny, což uživateli ušetří hledání a instalaci chybějících balíčků. Je zde umístěn i balíček `Diplomka.sty`, který zajišťuje vzhled titulní strany práce. Titulní strana práce je sázena podle zvyklostí ČVUT. Dále zde najdete balíček `Skripta.sty`, který definuje vzhled celé práce.

- Soubory:

Adresář Diplomka obsahuje hlavní zdrojový soubor `Diplomka.tex` a výsledné vzorové dokumenty `Diplomka.dvi`, `Diplomka.ps` a `Diplomka.pdf`.

Šablona je připravena pro výstup do `*.dvi`, nebo přes soubor `*.ps` do `*.pdf`. Lze překlad uskutečnit i pomocí programu `pdflatex.exe` přímo do `*.pdf`, ale v šabloně

<sup>3</sup><http://latex.feec.vutbr.cz/cz/latex/diskuze/>

<sup>4</sup><http://dce.felk.cvut.cz/roubal/teaching/latex.php>

pro tento druh překladu není připraven obrázek s logem ČVUT ve vhodném formátu. Čeština je v šabloně řešena přes balíček `babel` s parametrem `czech`.

Výhodou této šablony je jistě adresář `Styles`, který obsahuje nestandardní použité balíky. Uživatel je nemusí vyhledávat a instalovat. Dále se mi líbí rozdělení balíčku pro vzhled práce na dva soubory (`Diplomka.sty`, `Skripta.sty`), kde je v jednom definován vzhled titulní strany a ve druhém vzhled zbývající části práce. Lze tedy jednodušeji modifikovat titulní stranu, příp. použít tento balíček při sazbě práce pomocí jiné šablony, nebo úplně bez šablony.

U šablony `Diplomka` postrádám vhodnou dokumentaci, ve které by byly popsány zvolené parametry v preambuli zdrojového souboru. Nelíbí se mi, že není aktivováno záhlaví, a že při sazbě názvu kapitoly je uveden text *Kapitola n*, kde *n* je číslo kapitoly.

### 2.2.2 Šablona `Diplomka2`

Šablona `Diplomka2` vznikla přepracováním šablony `Diplomka`. Změny provedl Viktor Plaček. Tato šablona mě zaujala o něco více, než její původní verze, i když změny nejsou nikterak zásadní.

Oproti původní verzi přibylo záhlaví, kde je vypsána aktuální kapitola a číslo stránky. Záhlaví ale není odděleno od běžného textu žádnou linkou, což mi přijde nevhodné. Obsah adresáře `Styles` je totožný.

### 2.2.3 Další šablony

Z výše uvedených internetových stránek je možné získat také šablony pro psaní běžných úloh během studia a šablonu pro disertační práci.

- Šablony pro úlohy:

Jsou zde k dispozici dvě šablony pro běžné úlohy nebo semestrální práce. Obě jsou moc pěkné a jistě použitelné i pro naši fakultu. Úvodní stránka je odlišná.

- `uloha_LaTeX`: Jednoduchá šablona pro psaní úloh – Všechny definice i vlastní text obsahuje jeden soubor. Na začátku souboru jsou definovány okraje a lze je jednoduše přizpůsobit požadavkům autora. Text je rozdělen do 4 základních kapitol: *Zadání*, *Vypracování*, *Závěr* a *Reference*. Čeština je řešena přes balík `babel`. Zdrojový soubor je připraven pro překlad do `*.dvi`.
- `Vzor_TeX`: Vzorový dokument pro psaní referátů – Dokument je rozdělen do několika samostatných souborů podle kapitol a hlavní soubor

Vzor\_TeX.tex. Vzhled práce a sazba českých znaků jsou zajištěny balíčkem SAM. Zdrojový soubor je připraven pro překlad do souboru \*.dvi. Pro češtinu je použit styl czech. Obsahem dokumentu je návod, jak správně psát referát, aby byl výstižný, pochopitelný a poutavý.

- Šablona pro disertační práci:

Šablona pro disertační práci je funkční. Lze ji přeložit do \*.dvi i \*.pdf. Problémy se sázením českých znaků neřeší, je psaná v angličtině. Obsahuje adresář Styles a adresáře s jednotlivými kapitolami. Do hlubšího zkoumání šablony jsem se nepouštěla, ale myslím si, že je vhodné se o ní alespoň zmínit.

## 2.3 Šablona VŠB – TU Ostrava

Tato, dle mého názoru, moc pěkná šablona byla vytvořena na VŠB – TU Ostrava na Fakultě elektrotechniky a informatiky Jiřím Dvorským. Je k dispozici na stránkách: Typografický systém LaTeX<sup>5</sup>, kde lze najít i další užitečné informace a odkazy k L<sup>A</sup>T<sub>E</sub>Xu.

Šablona je vytvořena jiným způsobem než výše uvedené šablony. Makra, která jsou využívána při sazbě dokumentu, jsou definována ve vlastní třídě diploma.cls. V L<sup>A</sup>T<sub>E</sub>Xu je standardně na výběr několik tříd např. book, article, report. Aby byl vzhled dokumentu ovlivněn třídou, musí být zadána v preambuli dokumentu. Pro zadání třídy diploma bude tedy první řádek zdrojového souboru vypadat takto:

```
\documentclass[volby]{diploma}.
```

Jako nepovinný parametr při deklaraci třídy diploma se zadávají volby uvedené v dokumentaci k této třídě. Volby se týkají např. typu práce (bc – bakalářská práce, ing – magisterská práce), nastavení katedry (pro každou katedru je definován vlastní kód), pohlaví autora (male, female – použije se při sazbě automaticky vkládaných textů typu poděkování a prohlášení), . . .

### 2.3.1 Struktura šablony

Obsahem archívu diploma.zip, který je k dispozici na výše uvedených stránkách, jsou tyto soubory:

- diploma.cls = třída diploma – V tomto souboru jsou uložena makra pro sázení studentských prací. Je jich mnoho, některá z nich uvádí následující výčet, všechna jsou uvedena v dokumentaci k této třídě.

<sup>5</sup><http://www.cs.vsb.cz/dvorsky/Latex.html>

- Příkazy určené k zadávání údajů pro vysázení úvodních stran:
  - \* `\Author{}` – jméno autora
  - \* `\Title{}` – název práce
  - \* `\SubmissionDate{}` – datum odevzdání (např. 19. prosince 2008)
  - \* `\Thanks{}` – poděkování
  - \* `\CzechAbstract{}` – abstrakt česky
- Příkazy pro sazbu:
  - \* `\MakeTitlePages` – vysází úvodní stránky
  - \* `\InsertFigure` – vložení obrázku – Tento příkaz má 4 parametry. Velmi zjednoduší vkládání obrázku, syntax není složitá si zapamatovat:
 
$$\backslash\text{InsertFigure}\{soubor\}\{šířka\}\{název\}\{label\},$$
 kde *soubor* = cesta k souboru, kde je uložen obrázek, *šířka* = šířka obrázku, *název* = název obrázku, který se vysází pod ním, *label* = klíč ke křížovému odkazu, používá se při odkazech na obrázek v textu.
  - \* `\InsertSidewaysFigure` – vložení obrázku otočeného o 90°, stejné parametry jako `\InsertFigure`
- Makra pro sazbu zdrojových kódů pro různé programovací jazyky: Tato makra jsou propracována do detailů. Je možné vkládat zdrojové kódy přímo z textových souborů. Lze také vysázet seznam zdrojových kódů.
- Další makra: Jsou zde definována prostředí pro vytváření vět, definic a příkladů.

- `diploma.pdf` = dokumentace ke třídě `diploma` [3] – Dokumentace je vypracovaná výborně. Obsahuje všechny informace, které potřebuje běžný uživatel znát, aby mohl napsat práci pomocí této třídy. Dokumentace neobsahuje návod na instalaci  $\text{\LaTeX}$  a není učebnicí  $\text{\LaTeX}$ . Proto je třeba, aby uživatel měl nainstalovaný  $\text{\LaTeX}$  a uměl v něm vytvořit jednoduchý dokument před tím, než s touto třídou začne pracovat.

Dokumentace obsahuje i popis a definice jednotlivých maker. Ale není nutné se tím zabývat, pokud hodláte pouze napsat práci a nezajímá vás, jak celá šablona funguje.

- `Ukazka.tex/Ukazka.pdf` = vzorový dokument – Soubor `Ukazka.pdf` je výsledným dokumentem ze zdrojového souboru `Ukazka.tex`. Zdrojový soubor je napsán pro použití balíčku `czech` a tedy pro překlad pomocí programu `pdfcslatex.exe`. Podářilo se mi napoprvé soubor úspěšně přeložit. Je psaný

jednoduše a srozumitelně s vhodnými komentáři. Jistě ho lze použít jako výchozí soubor pro psaní studentské práce.

- `UkazkaBabel.tex/UkazkaBabel.pdf` = vzorový dokument – Pro soubory `UkazkaBabel.*` platí obdobný popis jako pro soubory `Ukazka.*` s tím rozdílem, že `UkazkaBabel.tex` využívá k sázení českých znaků balíček `babel` s parametrem `czech`, tudíž musí být přeložen programem `pdflatex.exe`. Problémem při použití tohoto typu češtiny je to, že v matematickém prostředí nefungují české akcenty.
- Ostatní soubory = různé soubory obsahující např. obrázky a grafy, které jsou vloženy do ukázkových souborů, dále soubory `*.log`, `*.lof`, ...

### 2.3.2 Zhodnocení třídy dokumentu diploma

Celkově mě třída `diploma` nadchla. Její použití je jednoduché, má srozumitelně psanou dokumentaci a výsledný dokument vypadá pěkně. Což jsou vlastnosti pro běžného uživatele nedocenitelné. Dalším pozitivem jsou makra vytvořená pro vkládání zdrojových kódů různých programovacích jazyků, bez kterých se většina studentských prací z technických škol neobejde.

Třída `diploma` je připravena pro přímý překlad dokumentu do `*.pdf`, což může být v některých případech omezující. Úvodní strany práce jsou uzpůsobeny zvyklostem na VŠB – TU Ostrava, které se diametrálně liší od zvyklostí na ČVUT.

## 3 Vytvoření šablony pro FSv ČVUT

Studentské práce na naší fakultě jsou velmi různorodé. Je to způsobeno tím, že studenti mají při tvorbě prací volnou ruku. Na internetových stránkách Fakulty stavební ČVUT se mi nepodařilo najít šablonu pro psaní studentských prací. Nenašla jsem ani pokyny, jak by měla práce vypadat. Myslím si, že je vhodné, aby práce studentů technicky zaměřené školy vypadaly profesionálně a měly jednotnou podobu. Snad k tomu přispěje i následující šablona. K vytvoření šablony jsem využila několik pěkně zpracovaných studentských prací jako vzor a Šablonu 2.02.

Nejprve bylo nutné zprovoznit Šablonu 2.02. Tato šablona má pěkně vypracovanou dokumentaci. Jsou v ní popsány zásadní problémy, které mohou při práci s šablonou nastat, včetně jejich řešení. Bylo třeba doinstalovat některé balíky, a pak už vše fungovalo dobře. Soubor `thesis.sty`, `path.sty` a `suffix.sty` jsem nakočkovala do adresáře `C:/texmf-local`, který je určený pro přidávání balíčků, které nejsou součástí instalace. Z instalace TeXLive jsem přidala balíček `acronym.sty`.

Poté jsem si prohlédla soubor `thesis.sty`. Tento soubor je vzhledem k univerzálnosti šablony poměrně rozsáhlý, což běžnému uživateli znesnadňuje orientaci v něm. Jsou zde uloženy definice různých příkazů, definice stylu `vutstyle`, vzhled úvodních stránek, seznamů zkratk, obrázků a tabulek, dále několik podmínek pro automatické změny vzhledu podle uživatelem navolených a zadaných informací. V tomto souboru jsem provedla několik úprav tak, jak to uvádějí následující kapitoly a přejmenovala jsem ho na `thesiscvut.sty`, aby nemohlo dojít k záměně s původním souborem. V dalším textu uvádím změny, které jsem udělala v Šabloně 2.02, abych ji přizpůsobila potřebám studentů na FSv ČVUT v Praze.

### 3.1 Obálka a titulní strana práce

Obálka a titulní strana práce jsou vysázeny pomocí příkazů `\vytvorobalku` a `\vytvortitulku`. Tyto příkazy jsou v původní šabloně připraveny tak, že vysází danou stránku s ohledem na to, jaký styl práce si uživatel vybral. Je to způsobeno tím, že oba tyto příkazy obsahují podmínky, které zohledňují volby uživatele. Všechny zadávané informace (jméno autora, název práce, ...) jsou ukládány do proměnných (`\autor`, `\nazev`, ...), pomocí nichž se potom sází uložený text na různá místa. Změny, které jsem v šabloně provedla se týkají pouze případu, kdy je zapnutý styl `cvutstyle` (tudíž je vypnuto sázení v angličtině). Styl `cvutstyle` jsem vytvořila přejmenováním a úpravou původního stylu `vutstyle`. Vložení správného obrázku při změně kompilátoru je zajištěno a samozřejmě je zohledněn i druh práce.

Při úpravě vzhledu úvodních stran jsem využívala následující možnosti:



- Stupeň písma:

V tabulce 3.1 jsou shrnuty možnosti nastavení stupně písma. Při úpravách úvodních stran šablony jsem používala stupeň `\normalsize` a stupně vyšší. Ostatní stupně uvádím, aby byl výčet možností kompletní.

| Příkaz                     | Velikost | Ukázka | Příkaz              | Velikost | Ukázka |
|----------------------------|----------|--------|---------------------|----------|--------|
| <code>\tiny</code>         | 5pt      | ABCdef | <code>\large</code> | 12pt     | ABCdef |
| <code>\scriptsize</code>   | 7pt      | ABCdef | <code>\Large</code> | 14,4pt   | ABCdef |
| <code>\footnotesize</code> | 8pt      | ABCdef | <code>\LARGE</code> | 17,28pt  | ABCdef |
| <code>\small</code>        | 9pt      | ABCdef | <code>\huge</code>  | 20,74pt  | ABCdef |
| <code>\normalsize</code>   | 10pt     | ABCdef | <code>\Huge</code>  | 24,88pt  | ABCdef |

Tab. 3.1: Možnosti nastavení stupně písma

- Zarovnání textu:

Zarovnávání textu lze v  $\text{\LaTeX}$ u zajistit použitím vhodného prostředí, nebo pomocí jednorázového příkazu. Při úpravách úvodních stran jsem používala příkazy pro zarovnání na střed a pro zarovnání vlevo. Pro prostředí pro zarovnávání textu jsou vhodná pokud je třeba zarovnat větší a ucelenou část textu.

- Příkazy:

- \* `\raggedright` – zarovnání vlevo
- \* `\raggedleft` – zarovnání vpravo
- \* `\centering` – zarovnání na střed

- Prostedí:

Prostředí vždy začíná příkazem `\begin{název prostředí}` a končí příkazem `\end{název prostředí}`. Veškerý text, který se nachází mezi těmito dvěma příkazy je ovlivněn zvoleným prostředím.

- \* `\begin{flushleft}`

Text, který je zarovnaný vlevo.

`\end{flushleft}`

- \* `\begin{flushright}`

Text, který je zarovnaný vpravo.

`\end{flushright}`

- \* `\begin{center}`

Text, který je zarovnaný na střed.

`\end{center}`

- Mezery:

Při vytváření textu, který není standardně uspořádán do odstavců, je třeba vkládat různě velké mezery v horizontálním i vertikálním směru. V  $\text{\LaTeX}$ u je pro tyto případy vytvořeno několik příkazů. Při úpravách úvodních stran šablony mi stačily příkazy:

$\text{\vfill}$  – maximální možná vertikální mezera vzhledem k okolnímu textu,

$\text{\hfill}$  – maximální možná horizontální mezera vzhledem k okolnímu textu.

Další možnosti vytváření vertikálních a horizontálních mezer jsou velmi přehledně uspořádány v Tematickém slovníčku v [10] na stranách 224 – 225.

- Změna měřítka obrázku:

Potřebovala jsem upravit velikost loga ČVUT. Použila jsem k tomu příkaz  $\text{\scalebox}$ . Syntax příkazu je:

$$\text{\scalebox{měřítko}{objekt}}.$$

Zdrojové kódy potřebné k vysázení úvodních stran jsou uvedeny v příloze A.1. Některé části jsou doplněny komentářem.

## 3.2 Prohlášení

Změny, které jsem provedla v prohlášení se týkaly pouze vlastního textu. Původní text byl úplně jiný, než je zvykem na naší fakultě. Při přípravě textů, které se vysází při použití šablony automaticky, je vhodné využít možností  $\text{\LaTeX}$ u pro zapisování akcentů nad latinková písmena. Přehled některých akcentů zapsaných pomocí standardních znaků shrnuje tabulka 3.2. Tento způsob vytváření akcentů jsem zvolila proto, že uživatelé šablony mohou používat různá kódování a české znaky by se jim nemusely zobrazovat správně.

|         |              |                |                |              |                |              |
|---------|--------------|----------------|----------------|--------------|----------------|--------------|
| příkaz  | $\text{\'e}$ | $\text{\v{c}}$ | $\text{\r{u}}$ | $\text{\"u}$ | $\text{\H{u}}$ | $\text{\.o}$ |
| výledek | é            | č              | ů              | ü            | ű              | ó            |

Tab. 3.2: Akcenty zapsané pomocí univerzálních znaků

## 3.3 Záhloví a zápatí

Záhloví jsem v původní šabloně vůbec neobjevila. Přidala jsem tedy další balíček `fancyhdr`, který umožňuje pracovat se záhlavím a zápatím. Tento balíček má dokumentaci v angličtině, která je součástí instalace TeXLive 2007 [8]. Na základě

vzorových diplomových prací a vlastním úsudku jsem určila vzhled záhlaví tak, že v levém horním rohu bude umístěno logo ČVUT a nápis ČVUT Praha, v pravém horním rohu bude číslo a název aktuální kapitoly.

- Levé záhlaví – Logo:

Pro vyplnění levého záhlaví jsem použila příkaz `\lhead{}`. Pomocí tzv. parboxu jsem umístila logo ČVUT i text ČVUT Praha. Obsah tohoto boxu je zpracováván v horizontálním režimu. Aby byl text zalamován v určitém místě, je nutné zadat povinný parametr – šířka = délka řádku. Parbox se vytvoří příkazem `\parbox`, který má tuto syntax:

$$\backslash\text{parbox}[umístění][výška][vnitřní\ pozice]\{šířka\}\{obsah\}$$

Více informací o parboxech naleznete v [10].

Do parboxu jsem umístila text ČVUT Praha i logo. Pro vložení obrázku jsem použila příkaz:

$$\backslash\text{includegraphics}[scale=1]\{\backslash\text{logo@zahl}\}.$$

Tento příkaz vyžaduje připojení balíku `graphics.sty` v hlavičce hlavního souboru `sablona.tex` pomocí `\usepackage{graphics}`. Vložení obrázku do záhlaví (ale i do ostatních částí dokumentu, které jsou připraveny předem v šabloně) bylo pro mě poměrně problematické. Bylo třeba vyřešit kolizi formátu obrázku se zvoleným kompilátorem. Pro výstup v `*.pdf` je možné vkládat obrázky v `*.pdf` a pro výstup v `*.dvi` je nutné vložit obrázek ve formátu `*.eps`. Proto bylo třeba vytvořit v souboru `thesiscvut.sty` proměnnou `\logo@zahl`. Do této proměnné se vloží cesta k obrázku ve správném formátu podle zvoleného kompilátoru. Využila jsem již definovanou strukturu příkazů, která je určena pro vkládání loga školy na titulní stránku. Obrázek musí být připraven v adresáři `obrazky` ve dvou formátech (`*.pdf`, `*.eps`).

- Pravé záhlaví – Název kapitoly

Zobrazení názvu kapitoly v pravém záhlaví je zajištěno použitím příkazu:

$$\backslash\text{pagestyle}\{fancy\}.$$

Pomocí tohoto příkazu se zavolá funkce `\chaptermark`, která implicitně zobrazí `1.KAPITOLA NÁZEV`. Vzhledem k možnosti delších názvů kapitol, jsem se rozhodla, že by bylo lepší, kdyby se zobrazoval název kapitoly takto `1.NÁZEV`. Bylo tedy třeba změnit příkaz `\chaptermark`. V dokumentaci k balíčku `fancyhdr.sty` [8] jsou popsány různé možnosti zobrazení názvu kapitoly.

Dalším úkolem bylo docílení toho, aby se v záhlaví zobrazovaly i názvy nečíslovaných kapitol. To jsem vyřešila přidáním příkazu:

$$\backslash\text{markboth}\{text\ do\ \text{záhlaví}\}\{\},$$

kde lze jako parametr zadat pouze text, který se má v záhlaví zobrazit, nebo zavolat příkazem nějakou proměnnou typu `\refname`. Poté příkaz vypadá takto:

$$\backslash\text{markboth}\{\backslash\text{MakeUppercase}\backslash\text{refname}\}\{\}.$$

Příkaz `\MakeUppercase\}` způsobí, že se text vysází velkými písmeny.

Musela jsem znovu definovat obsah zápatí, protože mojí definicí záhlaví se změnilo. Inspirovala jsem se Šablonou 2.02 (číslo stránky uprostřed).

Všechny zdrojové kódy potřebné k definici záhlaví jsou uvedeny v příloze A.2.

### 3.4 Vytvoření příkazu pro vysázení měsíce obhajoby

Součástí úvodních stran práce je na Fakultě stavební ČVUT zvykem uvádět měsíc obhajoby. V původní šabloně je vytvořen příkaz pouze pro rok a místo obhajoby. Proto jsem vytvořila vlastní příkaz, pro vysázení měsíce obhajoby.

K vytvoření příkazu `\mesic\}` jsem jako vzor použila příkaz pro zadání roku obhajoby. Jedná se o příkaz s jedním parametrem. Specialitou Šablony 2.20 jsou různá varovná hlášení. Pokud uživatel nezadá nějakou z podstatných informací, zobrazí se mu varování: „Nezadan ...“ v tomto případě : „Nezadan mesic“.

Zdrojové kódy potřebné k objasnění příkazu `\mesic\}` jsou uvedeny v příloze A.3.

## 4 Prezence pro publikum

Pokud je třeba vytvořit poutavou prezentaci pro představení své práce, lze k tomu použít kromě komerčních programů také nástroje L<sup>A</sup>T<sub>E</sub>Xu. Pro tvorbu prezentací jsou k dispozici různé balíky a třídy. Nejznámější, a v dnešní době pravděpodobně nepoužívanější třídou, je třída Beamer, která je součástí instalace TeXLive 2007.

### 4.1 Beamer

Beamer umožňuje vytvářet pěkné prezentace pomocí systému L<sup>A</sup>T<sub>E</sub>X poměrně snadno. Součástí této třídy je několik připravených témat a vzorové soubory pro různé příležitosti. Dokumentace je sice psaná v angličtině, ale je srozumitelná a přehledná, i když poměrně rozsáhlá. Na internetu lze najít i české návody a postřehy uživatelů, které pomůžou při řešení některých problémů, použila jsem např. [13]. Ke snadnému vytváření prezentací je vhodné zorientovat se ve struktuře adresářů, dále vybrat vhodný vzorový soubor a téma a vytvořit vlastní snímky prezentace. Nakonec je nutné zdrojový soubor přeložit. Je možné získat prezentaci v \*.pdf (použitím příkazu `pdflatex.exe`) i v \*.dvi (použitím příkazu `latex.exe`) formátu.

Samozřejmě jsou k dispozici i nástroje pro vytváření vlastních témat a prezentací s vlastním vzhledem. Ale v tomto případě již není tak snadné prezentaci vytvořit. Myslím si, že pro běžného uživatele je výběr připravených témat dostačující.

Další text by měl být návodem pro začátečníka, jak prezentaci pomocí třídy Beamer a jejích připravených nástrojů vytvořit. Při psaní tohoto textu jsem čerpala částečně ze svých zkušeností, ale převážně z dokumentace [11].

#### 4.1.1 Instalace třídy a struktura adresářů

Nainstalování třídy Beamer je velmi jednoduché pokud používáte některou z distribucí L<sup>A</sup>T<sub>E</sub>Xu, která ho obsahuje. V případě použití TeXLive 2007 se postup neliší od přidávání balíčků pomocí průvodce (viz kapitola 1.1.2).

Po instalaci třídy Beamer se v TeXLive 2007 vytvoří několik adresářů. Následující dva adresáře jsou nejdůležitější, a proto se pokusím popsat jejich obsah.

- ... \TeXLive2007\textmf-dist\doc\latex\beamer\ obsahuje:
  - Adresář `doc`: Tento adresář obsahuje dokumentaci ve formátu \*.tex. Hlavním souborem je `beameruserguide.tex`. V ostatních souborech jsou uloženy zdrojové kódy pro jednotlivé kapitoly dokumentace, které se spuštěním překladač hlavního souboru připojí. Dohromady pak vytvoří podrobný popis třídy Beamer.

- Adresář **emacs**: Obsahuje jeden soubor – pro uživatele (jako jsem já) nepodstatný.
- Adresář **examples**: Již podle názvu adresáře je jasné, že obsahuje příklady – soubory `*.tex`. V každém příkladě je předveden nějaký nástroj, prostředí nebo téma, aby si uživatel udělal přehled o základních možnostech práce s třídou Beamer. Tyto soubory nejsou vhodné jako šablony k tvorbě vlastní prezentace.
- Adresář **lyx**: Obsahuje příklad vytvořený speciálně pro LyX.
- Adresář **solutions**: Tento adresář je pro běžného uživatele nejpodstatnější, obsahuje šablony pro vytváření prezentací. Šablony jsou rozděleny do tří adresářů podle příležitosti, ke které se prezentace vytváří. Každý adresář obsahuje šablonu v angličtině, v němčině pro standardní editory a totéž pro LyX.
  - \* Adresář **conference-talks**: Šablona v adresáři conference-talks je vhodná pro prezentace trvající přibližně 20 minut.
  - \* Adresář **generic-talks**: Šablona v adresáři generic-talks je určena na představení práce v rozmezí 15 – 45 minut.
  - \* Adresář **short-talks**: Šablona v adresáři short-talks je určena pro krátká sdělení přibližně na 2 minuty.

Rozdělení do těchto skupin je samozřejmě pouze orientační.

- `... \TeXLive2007 \texmf-dist \tex \latex \beamer \`:

Tento adresář obsahuje soubory `*.sty` a další podadresáře:

- Adresář **art**: Tento adresář je pro běžného uživatele nepodstatný, obsahuje několik souborů s ikonami Beameru.
- Adresář **emulation**: Obsahem tohoto adresáře jsou balíky užívané při importu již vytvořených dokumentů v jiných třídách (prospers, seminar, ...) do třídy Beamer a příklady, které tyto balíky využívají. Jsou zde definovány příkazy k převodu některých nástrojů původní třídy (prospers, seminar) do třídy Beamer.
- Adresář **multimedia**: Obsahuje balíčky pro vytváření různých animací. Podle dokumentace umožňují tyto balíčky dokonce vkládání videí a zvuků.
- Adresář **themes**: V tomto adresáři se nachází soubory, které ovlivňují vzhled vytvořené prezentace. Témata se dělí na pět druhů.

- \* Adresář **color**: Adresář **color** obsahuje 17 **\*.sty** souborů. Tyto soubory definují barevné složení prezentace. Je v nich detailně nastaveno jaká barva bude použita pro konkrétní objekty v prezentaci.
- \* Adresář **font**: Adresář **font** obsahuje 6 **\*.sty** souborů. Tyto soubory definují různé kombinace fontů pro prezentaci.
- \* Adresář **inner**: Adresář **inner** obsahuje 5 **\*.sty** souborů. Tyto soubory definují vzhled vnitřních elementů (číslované a nečíslované výčty, obsah, ...) prezentace. Například v prostředí **enumerate** (číslovaný výčet) řeší, zda bude za číslem vysázena tečka, nebo nebude. Naopak neřeší font, kterým bude číslo vysázeno.
- \* Adresář **outer**: Adresář **outer** obsahuje 9 **\*.sty** souborů. Tyto soubory definují vnější elementy prezentace. Ovlivňuje například to, zda a kde bude umístěna linka oddělující záhlaví a zápatí od ostatního textu, kde bude umístěno logo, ...
- \* Adresář **theme**: Adresář **theme** obsahuje mnoho témat, která definují celkový vzhled prezentace. Jedná se o různé kombinace stylů z výše jmenovaných adresářů. Některé vlastnosti tématu lze přenastavit vložením konkrétního balíku.

### 4.1.2 Čeština

Zavedení češtiny do prezentace není problém. Je však nutné zavést češtinu přes balík `babel` s parametrem `czech`. Tento balík se použije přidáním příkazu:

```
\usepackage[czech]{babel}
```

do hlavičky souboru (viz kapitola 1.1.3). Balíček `czech` nelze použít, protože je pro něj nutné mít i počeštěný překladač `pdfcslatex.exe` (`cslatex.exe`), který není kompatibilní s některými příkazy používané ve třídě Beamer. Pro Beamer je tedy nutné použít program `pdflatex.exe` (`latex.exe`).

### 4.1.3 Vzhled prezentace

Vzhled prezentace se ve třídě Beamer určuje nejjednodušeji pomocí témat. Vhodný výběr tématu je pro prezentaci velmi důležitý. Je třeba se zabývat tím, zda je zvolena vhodná kombinace barev, jestli je font dobře čitelný, co se zobrazuje v záhlaví a zápatí, kde je umístěno logo. Některá témata se hodí pro konference, kde přednášející vystupuje před cizími lidmi, jiná pro představení práce známým lidem. Je třeba se vyvarovat několika základních chyb, některé uvádím v následujícím výčtu.

- Pokud je barva pozadí podobná barvě textu, není text dobře čitelný.

- Barva pozadí by neměla být tmavá, je to nevhodné při tisku.
- Je třeba vybírat kombinace barev tak, aby nevznikl problém (např. s čitelností textu), kdyby se na projektoru změnil jejich odstín.
- Nedoporučuje se používat font Times Roman, který je původně určen k sazbě textu tak, aby se ho vešlo co nejvíce na co nejmenší plochu papíru (noviny) a tudíž není na prezentacích dobře čitelný.
- Není vhodné použít téma, které obsahuje na každém slidu jméno autora, když publikum zná přednášejícího.

#### 4.1.4 Předdefinované příkazy

Ve třídě Beamer je mnoho předdefinovaných příkazů, které pomohou vytvořit prezentaci. Některé z nich se uvádějí v preambuli, jiné až v těle zdrojového souboru.

- Příkazy v preambuli:

Tyto příkazy slouží k zadávání informací o autorovi, názvu prezentace, . . . Lze je obvykle ovlivnit dvěma základními parametry. Nepovinný parametr se uvádí do hranatých závorek, povinný do složených závorek.

- `\title [] {}` – Slouží k zadání názvu prezentace (např. Diplomová práce). Jako nepovinný parametr lze uvést zkrácený název, který se zobrazuje v záhlaví nebo zápatí slidů podle zvoleného tématu.
- `\subtitle [] {}` – Slouží k zadání vedlejšího názvu prezentace (např. Prezentací možnosti systému L<sup>A</sup>T<sub>E</sub>X). Opět je možné do hranatých závorek zadat zkrácený tvar.
- `\author [] {}` – Slouží k zadání jména a příjmení autora. V případě, že se na prezentované práci podílelo více autorů z různých institucí, lze zde využít možnosti „indexování“ jednotlivých autorů a následné přiřazení správné instituce k jejich jménům. Ukázka zdrojového kódu je uvedena v příloze B.1.1.
- `\date [] {}` – Slouží k zadání datumu představení prezentace publiku. Opět je možné do hranatých závorek zadat kratší verzi pro zobrazení v záhlaví a zápatí slidy. Do povinného parametru lze zadat kromě data také příležitost, ke které je prezentace vytvořena.
- `\institute [] {}` – Slouží k zadání instituce, ve které autor působí. Tento příkaz je spojen s příkazem `\author`. Ukázka zdrojového kódu je uvedena v příloze B.1.2.



- `\logo{}` – Slouží k vložení obrázku do proměnné `\logo`. Tento příkaz má pouze povinný parametr, do kterého může být umístěn text nebo obrázek. Pro vložení obrázku lze použít příkaz:

`\includegraphics[scale=1]{cesta k obrázku}`.

- `\AtBeginSubsection[]{}`  – Zajistí zobrazení daného slidu před každou sekcí, příp. podsekcí. Ukázka použití tohoto příkazu je uvedena v příloze B.1.3.

Výčet předdefinovaných příkazů, které se uvádí v preambuli jistě není úplný.

- Příkazy v těle dokumentu:

Tyto příkazy slouží k vysázení různých částí prezentace. Jsou předdefinovány a běžný uživatel nepotřebuje zasahovat do způsobu jejich sázení. Existuje jich velmi mnoho, uvádím pouze některé.

- `\titlepage` – Vysází úvodní slide. Použije informace uvedené v preambuli.
- `\tableofcontents` – Vysází obsah. Má několik volitelných parametrů.
  - \* `\tableofcontents[pausesections]` – Při zobrazení obsahu se dělají pauzy mezi sekcemi. Postupně se zobrazují jednotlivé sekce. Myslím, že je to vhodné pro představení práce. Ke každé sekci může autor sdělit několik úvodních informací.
  - \* `\tableofcontents[currentsections,currentsubsections]` – Tyto parametry zajistí zvýraznění aktuální sekce a podsekcce. Většinou se používají v případě, kdy je v preambuli uveden příkaz `\AtBeginSubsection[]{}` .
- `\part` – Založí novou část prezentace. Pro každou část je vytvořen vlastní obsah. Dělit prezentaci do částí bych doporučila v případě, že se bude používat na několika následujících přednáškách. Pro každou přednášku je pak vyhrazena jedna část a přednášející se může pohodlně vracet i k slidům z předcházejících přednášek.
- `\partpage` – Vysází úvodní stránku části.

### 4.1.5 Postup vytvoření prezentace

Jednoduchý návod, jak vytvořit prezentaci, lze shrnout do několika kroků.

1. Příprava adresáře:

Je vhodné vytvořit pro každou prezentaci vlastní adresář. Do tohoto adresáře je třeba zkopírovat připravenou šablonu, která je uložena v:

```
...\\TeXLive2007\\texmf-dist\\doc\\latex\\beamer\\solutions.
```

## 2. První překlad šablony:

Šablona je nastavena tak, že by měla jít přeložit. Proto je dobré to zkusit ještě před prvními úpravami. Chybové hlášky by se pak měly týkat pouze chybějících balíků, které lze jednoduše doinstalovat (viz kapitola 1.1.2). Nejčastější chybou by mohlo být to, že se zdrojový soubor překládá počestěným překladačem (`cslatex.exe`, `pdfcslatex.exe`). Je tedy nutné nastavit překlad standardním programem (`latex.exe`, `pdflatex.exe`).

## 3. Obsah a struktura prezentace:

Je třeba si rozvrhnout strukturu prezentace. V těle zdrojového souboru vytvořte jednotlivé kapitoly. Na začátek prezentace umístěte obsah. Pokuste se znovu soubor přeložit. Pokud vše funguje tak jak má, přejděte k dalšímu kroku.

## 4. Volba vhodného tématu:

Téma definuje celkový vzhled prezentace (barva, font, vzhled jednotlivých elementů, ...). Ve třídě Beamer jsou předdefinovaná témata uložena jako stylové soubory v adresáři k tomu určeném (viz kapitola 4.1.1). Do prezentace se vkládají v hlavičce zdrojového souboru níže uvedenými příkazy. Obecná syntax příkazu je: `\usetheme[volby]{jméno stylu}`. Příklady použití jednotlivých příkazů:

- Barva – `\usecolortheme{rose}`
- Font – `\usefonttheme[onlylarge]{structuresmallcapsserif}`
- Vnitřní elementy – `\useinnertheme{circles}`
- Vnější elementy – `\useoutertheme{sidebar}`
- Celkový vzhled prezentace – `\usetheme{Warsaw}`

## 5. Vytvoření vlastních slidů:

Dalším krokem je vytvoření vlastních slidů. K tomu je třeba použít následující strukturu příkazů:

```
\begin{frame} % začátek slidu
  \frametitle{Název slidu}
  \framesubtitle{Upřesnění názvu slidu}
```

```
Vlastní text, obrázky, grafy, ...
\end{frame}      % konec slidu
```

Ukázka jednoduchého zdrojového souboru je uvedena v příloze B.2.

#### 6. Test prezentace:

Nakonec je nutné prezentaci otestovat. Doporučuji testovat prezentaci po každé změně.

## 4.2 Další možnosti při tvorbě prezentací v L<sup>A</sup>T<sub>E</sub>Xu

Kromě používané třídy Beamer existují i jiné nástroje. Některé se postupně přestaly používat, jiné se dále rozvíjí. V dalším textu uvádím některé z nich pouze s minimem informací a odkazy, kde lze tyto informace rozšířit. Tento výčet s největší pravděpodobností není úplný.

### 4.2.1 Seminar

Třída Seminar se používá jako základ jiných tříd. V dnešní době již není považována za vhodný nástroj k tvorbě prezentací. Má pěkně vypracovanou dokumentaci v angličtině [12] z roku 1993 a je součástí TeXLive 2007. Možnosti této třídy nejsou pro dnešní požadavky na prezentace dostačující. Ale v době jejího vzniku musela být převratná. Dokáže používat barvy, umisťovat rámečky, měnit šířku čar, používat různé fonty a také vytvářet slidy na výšku i na šířku papíru.

Více informací o třídě Seminar je k dispozici na: CTAN: View package information<sup>1</sup>.

### 4.2.2 Prosper, ha-prosper

Třída Prosper je určena k vytváření slidů pro projektory. Je napsána na základě třídy Seminar. Dokáže vytvářet slidy s obdobným vzhledem jako např. PowerPoint. Doplnkem této třídy je balík `ha-prosper`, který rozšíří nástroje Prosperu o automatické generování obsahu na libovolné stránce, podporu poznámek a možnost vytvářet slidy na výšku papíru. Oba nástroje jsou součástí TeXLive 2007 a je k nim vypracovaná dokumentace v angličtině [4] [5].

Více informací o třídě Prosper a balíku `ha-prosper` je k dispozici na: CTAN: View package information<sup>2</sup>.

<sup>1</sup><http://tug.ctan.org/pkg/seminar>

<sup>2</sup><http://tug.ctan.org/pkg/prosper>

### 4.2.3 Powerdot

Powerdot je třída L<sup>A</sup>T<sub>E</sub>Xu určená pro jednoduché a rychlé vytváření prezentací profesionálního vzhledu. Obsahuje mnoho nástrojů které obohatí prezentaci např. několik předdefinovaných vzhledů prezentací, osobní poznámky nebo režim pro tvorbu letáků. Výstupním formátem může být `*.pdf`, `*.ps`, `*.dvi`. Třída Powerdot obsahuje nástroje pro vytváření vlastních stylů prezentací a lze s ní pracovat i pod LyXem. K Powerdotu je vypracovaná přehledná a podrobná dokumentace v angličtině [6].

Více informací o třídě Powerdot je k dispozici na: CTAN: View package information<sup>3</sup>.

---

<sup>3</sup><http://tug.ctan.org/pkg/powerdot>

## 5 Šablona pro prezentace na FSv ČVUT

Zpracování šablon ve třídě Beamer mi přijde zajímavé a účelné. Rozhodla jsem se vytvořit obdobnou šablonu pro prezentaci prací na ČVUT. Výhodou této šablony oproti standardním šablonám ve třídě Beamer by mělo být několik faktorů:

- jazyk – Šablona bude psaná česky. Komentáře k jednotlivým příkazům budou pro některé uživatele srozumitelnější.
- volitelné parametry – Parametry příkazů, které ovlivňují vzhled prezentace, budou vypsané a zakomentovány, uživatel vybere možnost, která mu nejvíce vyhovuje, odkomentováním dané volby.
- znaky ČVUT – V šabloně budou vyplněny informace o instituci a přidáno logo ČVUT ve vhodném formátu.

Šablona tedy nepřekvapí žádnými novými tématy a příkazy. Využívá již definované nástroje třídy Beamer. Dokumentaci jsem nevytvořila. Myslím si, že k pochopení šablony postačí komentáře u jednotlivých příkazů, v případě nejasností pak dokumentace ke třídě Beamer nebo tato diplomová práce.

Při tvorbě šablony jsem částečně postupovala podle kroků pro vytváření prezentace uvedených v dokumentaci [11] a shrnutých v kapitole 4.1.5 této diplomové práce. Jako výchozí šablonu jsem zvolila `generic-ornate-15min-45min.en.tex`. Nakopírovala jsem ji do pracovního adresáře a přejmenovala na `CVUT.tex`. Pro češtinu je nutné ve třídě Beamer použít balíček `babel`, překlad lze tedy provést pouze programem `pdflatex.exe`. Pečlivě jsem prozkoumala obsah šablony. Pomocí dokumentace [11] a opakovaných překladů zdrojového souboru s různými změnami jsem zjistila význam použitých příkazů. K některým příkazům jsem doplnila komentáře, příp. připravila seznam možných parametrů. Konkrétní postup hlavních kroků při přípravě šablony popisují následující odstavce.

### 5.1 Doplnění šablony

Do původní šablony bylo třeba doplnit několik informací:

- čeština – Pro psaní českých znaků a správné kódování češtiny jsem doplnila do hlavičky souboru příkazy:

```
\usepackage[czech]{babel},
\usepackage[cp1250]{inputenc}.
```

- příkaz `\title` – Doplnila jsem parametr příkazu tak, že uživatel odkomentováním zvolené možnosti vybere, zda se jedná o diplomovou, bakalářskou nebo semestrální práci.
- příkazy `\author`, `\institute` – Parametry příkazů se liší podle množství autorů. Vypsala jsem tři možné varianty: pro jednoho autora, pro dva autory ze stejné instituce a pro dva autory z různých institucí (viz. přílohy B.1.1, B.1.2). Do nepovinného parametru u příkazu definující instituci jsem vepsala zkratku ČVUT. Obsah tohoto parametru se vysází do záhlaví nebo zápatí prezentace.
- příkaz `\logo` – Pro vložení loga jsem použila příkaz:

$$\backslash\text{logo}\{\backslash\text{includegraphics}[\text{scale}=1]\{\text{logo.pdf}\}\},$$

který zajistí vložení loga ČVUT do prezentace. Umístění loga na slidu určuje zvolené téma. Při použití některých témat je třeba změnit velikost loga. Pro tyto případy je určen volitelný parametr `scale` – měřítko. Pokud se nastaví `scale=1`, je obrázek vložen ve své původní velikosti. Pokud se zadá vyšší (nižší) číslo, obrázek se zvětší (zmenší). Nedoporučuje se výrazně měnit velikost rastrového obrázku pomocí tohoto parametru. Pokud se obrázek zvětší několikanásobně, tak se jeho kvalita sníží.

## 5.2 Výběr tématu

Vzhled prezentace je velmi důležitý. Ve třídě Beamer ho lze nejjednodušeji změnit používáním témat, která jsou uložena v adresáři `themes` (viz. kapitola 4.1.1). Témata se vkládají do prezentace příkazem:

$$\backslash\text{usetheme}\{\textit{název tématu}\}.$$

Postupně jsem vyzkoušela všechna předdefinovaná témata. S některými z nich se mi nepodařilo překládání prezentace uskutečnit. Vybrala jsem tedy dvanáct témat, která byla funkční a něčím mě zaujala. Tato témata jsem vepsala do parametru příkazu `\usetheme` a doplnila komentáři.

## 6 Webové stránky

V dnešní době je obvyklé zveřejňovat studentské práce na internetu. V  $\text{\LaTeX}$ u existuje několik nástrojů pro automatický převod textu do `html`, příp. `xhtml`. Tyto nástroje můžou studentovi velmi usnadnit práci s vytvářením stránek. Pokusila jsem se některé z nich prozkoumat. Nepodařilo se mi najít žádný jednoduchý, stručný návod, podle kterého by se mi podařilo stránky vytvořit. Dokumentace k jednotlivým balíkům jsou psané pouze v angličtině. Jsou poměrně obsáhlé a složité.

Překvapilo mě, kolik různých balíků pro převod textu z  $\text{\LaTeX}$ u do `html` existuje. Za všechny uvádím ty, které obsahuje TeXLive 2007 a několik dalších, které jsou zmiňovány v různých diskuzích. Popis jednotlivých balíků zde neuvádím. Je k dispozici na internetových stránkách <http://tug.ctan.org/pkg/>.

- TeXLive 2007
  - `tex4ht`
  - `latex2html`
- Ostatní balíky
  - `tth`
  - `tex2page`

### 6.1 Pokusy vytvoření internetových stránek

V této kapitole uvádím jednotlivé pokusy, které jsem s převodem z  $\text{\LaTeX}$ u do `html` provedla. Vytvořila jsem vzorový dokument `pokus.tex`, který obsahoval český text, matematický vzorec a obrázek. Pro sázení češtiny jsem použila balíček `czech.sty`. Vzorový dokument byl připraven pro překlad pomocí programu `pdfcslatex.exe`.

Na konci každé podkapitoly je uvedeno, co jsem daným pokusem zjistila. Některé pokusy vedly ke změnám ve vzorovém dokumentu.

#### 6.1.1 Editory

Všechny editory uvedené v kapitole 1.2 mají k dispozici nástroj pro tvorbu internetových stránek. Dokumentace k editorům ovšem nepopisují nastavení tohoto nástroje. Pokusila jsem se konverzi provést bez jakéhokoliv nastavení, pouhým spuštěním nástroje na vzorový dokument.

Zjištění:

1. Použití nástrojů k převodu textu z  $\text{\LaTeX}$  do `html`, které jsou součástí editorů pro vytváření textů v  $\text{\LaTeX}$ , není zatím náležitě popsáno v dokumentacích. Nepodařilo se mi je ke konverzi použít.
2. Nejlépe se mi jeví editor LyX, ve kterém se stránky vytvořily. Problémem je ovšem převod obrázků a matematických vzorců. Ve výsledných stránkách se nezobrazí.
3. Čeština se v LyXu zobrazuje správně.

Důsledek: K volání příkazů pro provedení konverze je nutné použít příkazový řádek.

### 6.1.2 Příkazový řádek

Pokud chceme použít příkazový řádek, tak musíme vědět, jaký příkaz voláme. Podle množství informací o jednotlivých balících na internetu jsem se rozhodla použít balíček `tex4ht` tzn. volání programu `htlatex.exe`. Při úpravě a překladu vzorového dokumentu jsem postupovala podle návodu na stránkách: [LaTeX pro web](#)<sup>1</sup>, kde je napsáno, že pro správnou funkci balíčku musí být zdrojový soubor přeložitelný do `*.dvi`. Upravila jsem tedy vzorový dokument tak, že jsem vkládaný obrázek převedla do formátu `*.eps`. Dále jsem do hlavičky souboru uvedla posloupnost příkazů:

```
\newif\ifweb
\ifx\ifHtml\undefined % Mimo HTML.
    \webfalse
\else % V HTML.
    \webtrue
\fi
```

Tato posloupnost příkazů zajistí univerzálnost zdrojového souboru pro překlad do `*.dvi` i do `html`. Překlad vzorového souboru jsem uskutečnila příkazem:

```
htlatex pokus.tex.
```

Po prvním překladu se vytvořil `pokus.html`. Po jeho otevření jsem zjistila, že se nezobrazí obrázek a vzorec. Informace v příkazovém řádku hlásí několik chyb. Z těchto hlášení je patrné, že posloupnost příkazů, které se provedou po spuštění programu `htlatex.exe`, je obsahem souboru `tex4ht.env`.

Zjištění:

---

<sup>1</sup><http://www.fi.muni.cz/~xruzick7/latex-pro-web/>



1. Pro program `tex4ht` je vhodné ve zdrojovém souboru použít obrázky ve formátu `*.eps`.
2. Příkazový řádek neřeší problémy se zobrazením obrázků a vzorců.
3. Zdroj chyb je nutné hledat pomocí souboru `tex4ht.env`.

Důsledek: Zadala jsem text `tex4ht.env image` do vyhledávače a snažila se najít řešení.

### 6.1.3 ImageMagick

Na stránkách: `tex4ht` a `cestina`<sup>2</sup> jsou uvedeny změny, které bylo třeba v souboru `tex4ht.env` udělat, aby se vytvořily internetové stránky i s obrázky. Je zde doporučeno nastavit cestu k programu `convert.exe`, který je součástí programu **ImageMagick**. Program `convert.exe` slouží k převodu obrázků z `*.eps` do formátu `*.png`. **ImageMagick** jsem neměla nainstalován. Je volně k dispozici na stránkách ImageMagick<sup>3</sup>.

**ImageMagick** je souhrn programů. Umí pracovat s mnoha (více než 100) formáty obrázků. Dokáže obrázky převádět z různých formátů na jiné, otáčet je, měnit barvy, dělat výřezy, ... Umožňuje také do obrázků kreslit. Jeho výhodou je, že ho lze použít v příkazovém řádku. Konverzi obrázku provedeme příkazem:

```
convert obrazek.jpg obrazek.png
```

Tento příkaz má mnoho parametrů. Jejich výčet s vysvětlením je uveden na stránkách: `Imagemagick`<sup>4</sup>.

Zjištění: Je třeba mít nainstalovaný nějaký konvertor obrázků např. **ImageMagick**.

### 6.1.4 Soubor `tex4ht.env`

Tento soubor obsahuje posloupnost příkazů, které se vykonávají při spuštění programu `htlatex.exe`. Zaměříme se na konverzi obrázků a vzorců. Obrázky a vzorce se shodně konvertují do `.png` souborů a ty se následně zobrazují na html stránkách. Proto v následujícím textu budu pod pojem obrázky zahrnovat i vzorce.

Obrázky se nejprve převedou z formátu `*.eps` do formátu `*.ps` a následně do `*.png`. Název těchto souborů pro vzorový soubor je ve tvaru `zzpokusNx.*`, kde  $N$  je pořadové číslo obrázku. Po nainstalování **ImageMagicku** jsem spustila konverzi a obrázky se opět nevytvořily. V příkazovém řádku se zobrazilo chybové hlášení

<sup>2</sup><http://lists.felk.cvut.cz/pipermail/cstex/2006-May/020493.html>

<sup>3</sup><http://www.imagemagick.org/>

<sup>4</sup><http://www.imagemagick.org/script/convert.php>

### DVI file contains unexpected Omega command

Zadala jsem tuto hlášku do vyhledávače a na stránkách Dvips<sup>5</sup> jsem našla řešení. V souboru `tex4ht.env` jsem do příkazu, který je volán těsně před chybovou hláškou, doplnila parametr `-noomega0`. Dále jsem tento parametr doplnila všude, kde byl volán příkaz `dvips`. Výsledné příkazy v souboru `tex4ht.env` tedy vypadají takto:

```
Gdvips -Ppdf -noomega0 -mode ibmvga -D 110 -f
Gdvips -E -noomega0 -Ppdf -mode ibmvga -D 110 -f
```

Další pokusný překlad ukázal, že se obrázky opět nezobrazily. Ale vznikl soubor `zypokus.ps`. Chyba tedy musí být v převodu obrázku z `*.ps` do `*.png`. Našla jsem jednoduchý konvertor `ps2pdf.exe` a ověřila jsem, jestli v souboru `*.ps` je opravdu umístěn daný obrázek. Obrázek v souboru opravdu byl. Začala jsem tedy zkoumat chybové hlášení. Systém píše, že volá příkaz `convert` takto:

```
System call: c:\imagemagick\convert -trim +repage -density 110x110
-transparent "#FFFFFF" zypokus.ps zypokus.png pokusNx.png
Call failed (Error 5).
```

*N* je pořadové číslo obrázku. Po několika dalších marných pokusech jsem zjistila, že celý problém je v tom, že je nastavena špatně cesta k programu `convert`. Opravila jsem tedy cestu v souboru `tex4ht.env` na správnou a výsledný příkaz tedy vypadá takto:

```
c:\TeXLive2007\imagemagick\convert -trim +repage -density 110x110
-transparent "#FFFFFF" zz%%4.ps %%3
```

Další překlad vzorového souboru ukázal, že se obrázky konečně zobrazily.

Zjištění:

1. Je třeba doplnit parametr `-noomega0` k příkazu `dvips`.
2. Je třeba nastavit správně cestu k programu `convert.exe`.

## 6.2 tex4ht a šablona pro studentské práce

Dalším pokusem jsem chtěla zjistit, zda i složitější dokument půjde pomocí balíku `tex4ht` přeložit do `html`. Jako vzorový dokument jsem tedy použila šablonu pro psaní studentských prací ČVUT (soubor `sablona.tex`), jejíž vznik je popsán v kapitole 3.

<sup>5</sup><http://www.tug.org/texinfohtml/dvips.html>

Nastavila jsem šablonu tak, aby byla připravena na překlad do `*.dvi`. Udělala jsem testovací překlad a vše proběhlo bez chyby. Pomocí příkazového řádku a programu `htlatex.exe` jsem vytvořila soubor `sablona.html`. Během překladu proběhlo několik chybových hlášek. Týkaly se českých uvozovek. Ve výsledném souboru uvozovky nebyly zobrazeny. Úvodní strany se nevysázely správně. Zarovnání jednotlivých částí textu nebylo dodrženo. Háčky nad velkými písmeny v nadpisech se také nevysázely správně.

Úvodní strany jsou v šabloně složitě definovány. Proto si myslím, že pro konverzi do `html` by bylo vhodné je vytvořit zvlášť a jednodušeji. Nebudu se tím tedy dále zabývat.

Domnívala jsem se, že by špatné sázení českých znaků mohlo být způsobeno použitím balíčku `czech` pro sazbu českých znaků. Zaměnila jsem tedy tento balíček za univerzálnější `babel`. Ve výsledném dokumentu se ale nic nezměnilo. Pokusila jsem se tedy napsat akcentovaná písmena v nadpisech pomocí standardních znaků (viz. kapitola 3.2). Také to nepřineslo žádnou změnu ve výsledku.

Další kroky vedly ke změnám ve stylu, který ovlivňuje vzhled práce. Změnila jsem tedy v souboru `thesiscvut.sty` (řádek 488) definici názvu kapitoly tak, aby se nesázela velkými písmeny. Pokud se název kapitoly napíše do zdrojového souboru velkými písmeny, vysází se akcenty správně.

```
% pro konverzi do html
\normalfont\Large\bfseries

% pro sazeni kapitol velkými písmeny
\normalfont\Large\bfseries\MakeUppercase
```

Po této úpravě se vysázely názvy kapitol správně. Nechala jsem tedy v souboru `thesiscvut.sty` obě varianty s komentáři, aby si případný zájemce mohl vybrat možnost, která mu více vyhovuje. Problém se sázením českých uvozovek se mi nepodařilo vyřešit vůbec.

## 6.3 Shrnutí

Podařilo se mi vytvořit jednoduché internetové stránky pomocí nástroje `tex4ht`, který je součástí instalace TeXLive 2007. V předcházejících kapitolách je uveden postup, kterým jsem toho docílila, krok po kroku i s různými nezdary, což může čtenáře rozptylovat od podstaty textu. Proto se pokusím ve stručnosti shrnout důležité kroky v následujícím výčtu.

1. Instalace balíku `tex4ht` – Podle standardního postupu přidávání balíků je třeba přidat balík `tex4ht`.

2. Instalace vhodného konvertoru obrázků – Balík `tex4ht` neumí převádět obrázky a vzorce do formátu `*.png`, je tedy nutné nějaký konvertor nainstalovat. Vhodným kandidátem je program **ImageMagick**. Je volně dostupný na internetu a lze ho bez problémů použít v prostředí příkazového řádku.
3. Úprava zdrojového souboru pro překlad do `html` – Před tím, než se začne převádět zdrojový soubor na internetové stránky, je nutné přidat do hlavičky tohoto souboru posloupnost příkazů uvedenou v kapitole 6.1.2. Slova, která se mají vysázet velkými písmeny pomocí příkazu `\MakeUppercase`, je vhodné přepsat pomocí velkých písmen již ve zdrojovém dokumentu. Řešení pro vysázení českých uvozovek se mi nepodařilo najít vůbec. Další podmínkou pro správné vytvoření stránek je to, že lze zdrojový soubor přeložit do formátu `*.dvi`.
4. Změny v souboru `tex4ht.env` – Tento soubor obsahuje posloupnost příkazů, které se provedou při konverzi. Je nezbytné zkontrolovat, případně nastavit absolutní cestu k programu `convert.exe`, který je umístěn v **ImageMagicku**. Pokud není cesta správně nastavena, nevytvoří se ve výsledném `html` souboru obrázky a vzorce. Další změna v souboru `tex4ht.env` se týká volání programu `dvips.exe`. Pro správnou funkci převodu je zde nutné přidat parametr `-noomega0`.
5. Příkazový řádek – V příkazovém řádku je nutné vyhledat adresář, kde je umístěn zdrojový soubor (např. `pokus.tex`). Příkazem `htlatex pokus.tex` se spustí konverze. Vytvoří se výsledný soubor `pokus.html`. Příkaz `htlatex` disponuje různými možnostmi, které lze nastavit pomocí parametrů.
6. Šablona pro psaní studentských prací – Nepodařilo se mi úspěšně přeložit šablonu pomocí nástroje `tex4ht`. Vzhled úvodních stránek není absolutně dodržen. Některé předdefinované funkce šablony nejsou slučitelné s převodem do `html` formátu.

## Závěr

Cílem této diplomové práce bylo seznámit čtenáře s prezentačními možnostmi systému  $\LaTeX$ . Systém  $\LaTeX$  disponuje nástroji, které lze využít pro tvorbu typograficky velmi kvalitních dokumentů (s možností vkládání vzorců, tabulek, schémat a obrázků), jednoduchých a vkusných prezentací i internetových stránek. Myslím si, že  $\LaTeX$  je vhodným nástrojem pro vytváření závěrečných prací na vysokých školách. Výsledné dokumenty vznikají v netextových formátech (`*.dvi`, `*.pdf`), což zaručí stejný vzhled práce na různých počítačích a vyloučí možnost chyby, která by mohla vzniknout při tisku. Systém  $\LaTeX$  je volně k dispozici na internetu včetně veškerých doplňkových programů (editory, ImageMagick – program pro převod obrázků do nejrůznějších formátů), návodů a balíků.

Práce byla zpracována v systému  $\LaTeX$ . K instalaci systému byla použita distribuce TeXLive 2007. Pro editaci textu jsem zvolila program Texmaker.

Jedním z výstupů této práce je šablona, která by měla pomoci studentům Geodézie a kartografie na FSv ČVUT v Praze s vytvořením závěrečné práce v  $\LaTeX$ u. Tato šablona byla vytvořena úpravou šablony, která je určena pro studenty VUT v Brně. Definiuje strukturu závěrečné práce a vzhled úvodních stran dokumentu, záhlaví, nadpisů, . . . Měla by také přispět k tomu, aby práce studentů byly vzájemně srovnatelné a byl určen vzhled alespoň úvodních stran. Šablona byla použita i při psaní této práce.

Dalším výstupem této práce je šablona pro vytváření prezentací vhodných k představení studentské práce. Je uzpůsobena pro potřeby studentů ČVUT. Byla vytvořena pomocí předdefinovaných nástrojů třídy Beamer. Obsahuje několik sekcí, ve kterých si uživatel, podle informací uvedených přímo v šabloně, zvolí vzhled výsledné prezentace. Nepřináší tedy nic nového, pouze shromažďuje informace a možnosti použité třídy, což výrazně usnadňuje práci.

Pro tvorbu internetových stránek v systému  $\LaTeX$  je připraveno mnoho nástrojů. Některé jsou uvedeny v poslední kapitole této práce. Podrobněji jsem se zabývala pouze nástrojem `tex4ht`. Pomocí tohoto nástroje se mi po několika nezdařených pokusech podařilo automaticky převést jednoduchý text s obrázkem a vzorcem do formátu `html`. S převodem složitějšího textu jsem si neporadila. Problémy jsou např. s českými uvozovkami.

Psaní této práce mi bylo velkým přínosem. S  $\LaTeX$ em jsem se dříve nikdy neseťkala, proto si myslím, že mé postřehy by mohly pomoci začátečníkům s překonáním počátečních obtíží při používání tohoto všestranného nástroje. Doufám, že šablony budou studentům dobře sloužit a přispějí k vytvoření kvalitních závěrečných prací a prezentací.

## Použité zdroje

- [1] BEDNÁŘ, R. *LaTeX manual* [online]. [cit. 2008-12-5]. Dostupné z URL: <<http://www.cstug.cz/latex/lm/frames.html>>.
- [2] BOLDIŠ, P. *Bibliografické citace dokumentů podle ČSN ISO 690 a ČSN ISO 690-2* [online]. 2001, poslední aktualizace 2004-11-11. [cit. 2008-11-8]. Dostupné z URL: <<http://www.boldis.cz/citace/citace.html>>.
- [3] DVORSKÝ, J. *diploma – třída dokumentů pro diplomové práce* [online]. 2008-5-3. [cit. 2008-11-11].  
Dostupné z URL: <<http://www.cs.vsb.cz/dvorsky/Latex.html>>.
- [4] GOUALARD, F. – NEERGAARD, P. M. *Making slides in LaTeX with prosper* [online]. [cit. 2008-11-25].  
Dostupné z URL: <<http://amath.colorado.edu/documentation/LaTeX/prosper/0/prosper-doc.pdf>>. Též součást instalace TeXLive 2007: `\TeXLive2007\texmf-dist\doc\latex\prosper`.
- [5] HENDRI, A. *HA-prosper package – Documentation – Version 4.21* [online]. 2005-9-3. [cit. 2008-12-1].  
Dostupné z URL: <<http://tug.ctan.org/tex-archive/macros/latex/contrib/ha-prosper/Doc/HA-prosper.pdf>>.
- [6] HENDRI, A. – CHRISTOPHER, E. *The powerdot class* [online]. 2005-12-6. [cit. 2008-11-13].  
Dostupné z URL: <<http://tug.ctan.org/tex-archive/macros/latex/contrib/powerdot/doc/powerdot.pdf>>.
- [7] MARTÍNEK, D. *LaTeXové speciality* [online]. Poslední aktualizace 2008-2-24. [cit. 2008-12-13]. Dostupné z URL: <<http://www.fit.vutbr.cz/martinek/latex/>>.
- [8] OOSTRUM, P. *Page layout in LaTeX* [online]. 2004-3-2. [cit. 2008-11-3].  
Dostupné z URL: <<http://www.ctan.org/tex-archive/macros/latex/contrib/fancyhdr/fancyhdr.pdf>>. Též součást instalace TeXLive 2007: `\TeXLive2007\texmf-dist\doc\latex\fancyhdr`.
- [9] ROUBAL, J. *Publikační systém LaTeX* [online]. Poslední aktualizace 2007-12-3. [cit. 2008-10-29]. Dostupné z URL: <<http://dce.felk.cvut.cz/roubal/teaching/latex.php>>.

- [10] RYBIČKA, J. *L<sup>A</sup>T<sub>E</sub>X pro začátečníky* 3. vydání. Brno: KONVOJ, 2003. 238 s. ISBN 80-7302-049-1.
- [11] TANTAU, T. *User's Guide to the Beamer Class, Version 3.06* [online]. [cit. 2008-11-23]. 2005-10-23. Dostupné z URL: <<http://latex-beamer.sourceforge.net>>. Též součást instalace TeXLive 2007: `\TeXLive2007\texmf-dist\doc\latex\beamer\doc`.
- [12] ZANDT, T. V. *A L<sup>A</sup>T<sub>E</sub>X style for slides and notes – User's Guide* [online]. 1993-4-1. [cit. 2008-11-24]. Dostupné z URL: <<http://tug.ctan.org/tex-archive/macros/latex/contrib/seminar/doc/sem-user.pdf>>.
- [13] ZELENKA, P. *Beamer: LaTeX na prezentace* [online]. Poslední aktualizace 2006-8-28. [cit. 2008-11-7]. Dostupné z URL: <<http://http://www.abclinuxu.cz/clanky/navody/beamer-latex-na-prezentace>>.

# Seznam obrázků

|      |   |    |
|------|---|----|
| 1.1  | Průvodce instalací TeXLive 2007 . . . . .                             | 4  |
| 1.2  | Prostředí pro přidávání a odebírání balíků pro TeXLive 2007 . . . . . | 5  |
| 1.3  | Texmaker – Základní prostředí . . . . .                               | 7  |
| 1.4  | Texmaker – Nastavení . . . . .  | 8  |
| 1.5  | TeXnicCenter – Základní prostředí . . . . .                           | 9  |
| 1.6  | TeXnicCenter – Nastavení formátu výstupního souboru . . . . .         | 10 |
| 1.7  | Winshell – Základní prostředí . . . . .                               | 11 |
| 1.8  | Winshell – Výřez z panelů nástrojů . . . . .                          | 11 |
| 1.9  | WinShell – Možnosti nastavení . . . . .                               | 12 |
| 1.10 | LyX – Základní prostředí . . . . .                                    | 13 |



# Seznam tabulek

|     |  |    |
|-----|--|----|
| 3.1 | Možnosti nastavení stupně písma . . . . .            | 23 |
| 3.2 | Akcenty zapsané pomocí univerzálních znaků . . . . . | 24 |

## A Změny v Šabloně 2.02

Soubor `thesiscvut.sty` vznikl ze souboru `thesis.sty` a obsahuje definice příkazů pro vzhled výsledné studentské práce. Soubor `sablona.sty` obsahuje strukturu výsledného dokumentu. Tyto dva soubory mají na sebe úzkou návaznost, proto bylo většinou nutné při změnách zasahovat do obou dvou.

### A.1 Obálka a titulní strana

Při změnách v sazbě obálky a titulní strany nebylo třeba zasahovat do hlavního souboru `sablona.tex`. Veškeré změny jsem tedy provedla v souboru `thesiscvut.sty`.

#### A.1.1 Příkaz pro vytvoření obálky

- Definice příkazu `\vytvorobalku`:

Tuto část zdrojového souboru uvádím, aby čtenář měl možnost poznat, jakým způsobem je upraveno sázení textu v závislosti na zvoleném stylu práce. Převzala jsem ji z původní šablony.

```
% Sazba kompletneho prebalu prace prikazem \vytvorobalku %
\newcommand\vytvorobalku{\par
\begingroup
\renewcommand\thefootnote{\@fnsymbol\c@footnote}%
\def\@makefnmark{\rlap{\@textsuperscript{
\normalfont\@thefnmark}}}%
\long\def\@makefntext##1{\parindent 1em\noindent
\hb@xt@1.8em{%
\hss\@textsuperscript{\normalfont\@thefnmark}}##1}%
\if@twocolumn
\ifnum \col@number=\@ne
\ifTT@cvutstyle
\@vytvorobalku@cvut
\else
\@vytvorobalku
\fi
\else
\twocolumn[
\ifTT@cvutstyle
\@vytvorobalku@cvut
```

```

        \else
        \@vytvorobalku
        \fi ]
    \fi
\else
    \newpage
    \global\@topnum\z@    % Prevents figures from
                        % going at top of page.

    \ifTT@cvutstyle
        \@vytvorobalku@cvut
    \else
        \@vytvorobalku
    \fi
\fi
\endgroup
%% Tady zrusime prikazy
\global\let\vytvorobalku\relax
\global\let\@vytvorobalku\relax
}

```

- Vlastní vzhled obálky

```

%definice vzhledu obalky
\def\@vytvorobalku@cvut{%
    \renewcommand{\baselinestretch}{1.0}
    \newpage
    \thispagestyle{empty}
    \noindent
    \raisebox{5mm}[8mm][0mm]{%
        \begin{minipage}[t][247mm]{\linewidth}%
            \noindent %
            \parbox % vytvoreni parboxu
                % sirka parboxu = aktualni sirka radku
                {\linewidth}{
                    \centering % zarovnani na stred
                    % nastaveni rodiny pisma na sans serif
                    \sffamily
                    \LARGE % definice velikosti pisma
                    % vysazi velkými písmeny název školy

```

```

\MakeUppercase{\skola@cz} \\
}

\parbox
% vertikalni umisteni parboxu na stred okolniho textu
[c]
[10mm] % vyska parboxu 10 mm
% umisteni obsahu parboxu na stred parboxu
[c]
% sirka parboxu = aktualni sirka radku
{\linewidth}{
\centering % zarovnani na stred
% nastaveni rodiny pisma na sans serif
\sffamily
\large % definice velikosti pisma
% vysazi velkymi pismeny nazev fkulty
\MakeUppercase{\fakulta@cz} \\
}

% maximalni mozna vertikalni mezera vzhledem k nastaveni stranky
\vfill
\parbox
% vertikalni umisteni parboxu na stred okolniho textu
[c]
[1cm] % vyska parboxu 1 cm
% zarovnani obsahu parboxu na horni okraj parboxu
[t]
% sirka parboxu = aktualni sirka radku
{\linewidth}{%
\centering %zarovnani na stred
% nastaveni rodiny pisma na sans serif
\sffamily
\huge % definice velikosti pisma
% vysazi velkymi pismeny oznaceni prace
\MakeUppercase{\oznaceni prace@cz} \\
}

% maximalni mozna vertikalni mezera vzhledem k nastaveni stranky
\vfill
\noindent %
\parbox
% vertikalni umisteni parboxu na horni okraj okolniho textu
[t]

```

```

    [5mm]% vyska parboxu 5 mm
    % zarovnani obsahu parboxu na horni okraj parboxu
    [t]
    % sirka parboxu = aktualni sirka radku
    {\linewidth}{%
    \raggedright %zarovnani doprava
    % nastaveni rodiny pisma na sans serif
    \sffamily
    \large {% definice velikosti pisma
% vysazi velkými písmeny místo a rok obhajoby
    \MakeUppercase{\@misto} {\@rok}
% maximalni možna horizontalni mezera vzhledem k šířce stránky
    \hfill
    {\@autor@jmeno} % jmeno autora
    %prijmeni autora velkými písmeny
    \MakeUppercase{\@autor@prijmeni}\\}
    }
\end{minipage}}

```

## A.1.2 Příkaz pro vytvoření titulní strany

- Definice příkazu `\vytvortitulku`:

Tuto část zdrojového souboru uvádím, aby čtenář měl možnost poznat, jakým způsobem je upraveno sázení textu v závislosti na zvoleném stylu práce. Převzala jsem ji z původní šablony.

```

% Sazba kompletní úvodní stránky příkazem \vytvortitulku %
\newcommand\vytvortitulku{\par
\begingroup
\renewcommand\thefootnote{\@fnsymbol\c@footnote}%
\def\@makefnmark{\rlap{\@textsuperscript{
\normalfont\@thefnmark}}}%
\long\def\@makefntext##1{\parindent 1em\noindent
\hb@xt@1.8em{%
\hss\@textsuperscript{
\normalfont\@thefnmark}}##1}%
\if@twocolumn
\ifnum \col@number=\@ne
\ifTT@cvutstyle

```

```

        \@vytvortitulku@cvut
    \else
        \@vytvortitulku
    \fi
\else
    \twocolumn[
    \ifTT@cvutstyle
        \@vytvortitulku@cvut
    \else
        \@vytvortitulku
    \fi]
\fi
\else
    \newpage
    \global\@topnum\z@
    \ifTT@cvutstyle
        \@vytvortitulku@cvut
    \else
        \@vytvortitulku
    \fi
\fi
\endgroup

```

- Vlastní vzhled titulní strany:

```

\def\@vytvortitulku@cvut{%
    \renewcommand\baselinestretch{1.0}
    \newpage
    \thispagestyle{empty}
    \noindent
    \raisebox{5mm}[42mm][0mm]{%
        \begin{minipage}[t][247mm]{\linewidth}%
            \noindent%
            \parbox[b][42mm][t]{\linewidth}{
                \centering
                \sffamily \LARGE
                \MakeUppercase{\skola@cz}\\
                \sffamily \Large
            }
        }
    }

```

```

        \MakeUppercase{\fakulta@cz}\\
        \MakeUppercase{\oborstudiatext@cz}
        \MakeUppercase{\oborstudia@cz}\\
    }
\parbox[b][55mm][b]{\linewidth}{%
    \centering
    \scalebox{1.30}{%
        \includegraphics{\logo@skola}
    }}
\parbox[t][75mm][c]{\linewidth}{%
    \centering
    \sffamily \LARGE
        \MakeUppercase{\oznaceni@prace@cz}\\
    \sffamily \Large
        \MakeUppercase{\nazev@cz}\\%
}
\vfill
\parbox[t][33mm][t]{\linewidth}{%
    \centering%
    \sffamily \large %
        \makebox{\garanttext@cz} {\@garant}\\
        \makebox{\ustav@cz}
}
\parbox[t][5mm][t]{\linewidth}{%
    \raggedright%
    \sffamily \large %
        \MakeLowercase{\@mesic} \@rok
        \hfill {\@autor@jmeno}
        \MakeUppercase{\@autor@prijmeni}
}
    \end{minipage}%
}
\newpage
}

```

## A.2 Řešení záhlaví

V následujících částech této přílohy jsou uvedeny části zdrojových souborů, které řeší vzhled záhlaví.

### A.2.1 Soubor `thesiscvut.sty`

- Vytvoření příkazu `\logo@zahl` s podmínkami pro volbu správného formátu obrázku podle zvoleného typu překladu:

```
\ifTT@dvi pdfm
  \newcommand\logo@zahl{obrazky/logo2.pdf}
\fi
\ifTT@pdftex
  \newcommand\logo@zahl{obrazky/logo2.pdf}
\fi
\ifTT@dvi ps
  \newcommand\logo@zahl{obrazky/logo2.eps}
\fi
```

- Vytvoření příkazu `\zahlavi`:

```
\def\zahlavi
  \pagestyle{fancy}
  \lhead{\parbox[b]{}[b]{0.8cm}{
    \includegraphics[scale=0.03]{\logo@zahl}}ČVUT Praha}
  %v zápatí bude uprostřed umístěno číslo stránky
  \cfoot{\thepage}
  %linka pro oddělení záhlaví od běžného textu
  \renewcommand{\headrulewidth}{0.4pt}}
```

### A.2.2 Soubor `sablona.tex`

- Příkaz pro aktivaci záhlaví:

```
\zahlavi
```

- Úprava příkazu `\chaptermark`:  
Implicitně způsobuje příkaz `\chaptermark` vysázení textu *1.KAPITOLA NÁZEV*. Tento styl jsem změnila na *1.NÁZEV* následujícími příkazy.



```

\renewcommand{\chaptermark}[1]{ } % úprava příkazu \chaptermark
\markboth{\MakeUppercase{
\thechapter.\# 1}}{}}

```

### A.2.3 Ostatní soubory

Změny v ostatních souborech byly nutné zejména kvůli zobrazení názvu kapitoly, která není číslována, v záhlaví. U jednotlivých kapitol tento problém vyžadoval různá řešení.

- `uvod.tex`

```

% * zajistí, aby nebyla kapitola očíslována
\chapter*{Úvod}
% zajistí, aby se text Úvod zobrazil v obsahu
\addcontentsline{toc}{chapter}{Úvod}
% zajistí, aby byl text ÚVOD v záhlaví
\markboth{ÚVOD}{}

```

- `zdroje.tex`

Aby se v záhlaví zobrazil text POUŽITÉ ZDROJE, musela jsem zasáhnout do souboru `thesiscvut.sty`, kde je tento seznam definován. V původní šabloně bylo definováno, aby se seznam zdrojů do záhlaví nezobrazoval. Výsledný zdrojový kód vypadá následovně:

```

% nejprve je definována proměnná \refname
\def\refname{Použité zdroje}

% Literatura bez zahlavi nebo se zahlavim
% změna definice prostředí thebibliography
\renewenvironment{thebibliography}[1]
% zajistí nečíslování kapitoly
{\chapter*{\refname}
% vloží text, který je uložen v proměnné \refname do záhlaví
\markboth{\MakeUppercase\refname}{}
% vloží text, který je uložen v proměnné \refname do obsahu
\addcontentsline{toc}{chapter}{\refname}

```

- `zkratky.tex`

Seznam symbolů, veličin a zkratk je obdobný případ jako Seznam použitých zdrojů.

```

% definice prostředí seznamzkratek
    \newenvironment{seznamzkratek}[1]{
% * nečíslovaná kapitola
    \chapter*{Seznam symbol\{r\{u\}, veli\{v\{c\}in a zkratek}
% zahrnutí do obsahu
    \addcontentsline{toc}{chapter}{Seznam ...}
%zahrnutí do záhlaví
    \markboth{\MakeUppercase {Seznam ...}}{}
%... dále pokračuje definice příkazy, které nejsou
%   předmětem této přílohy

```

## A.3 Měsíc obhajoby

Na Fakultě savební ČVUT je zvykem uvádět na titulní stránce měsíc obhajoby.

### A.3.1 Soubor thesiscvut.sty

```

% Definice mesice obhajoby
\def\mesic#1{\gdef\@mesic{#1}}
\def\@mesic{\@latex@warning@no@line{Nezadan mesic\noexpand\mesic}}

```

### A.3.2 Soubor sablona.tex

```

% Mesic obhajoby
\mesic{leden} % zadani mesice ledna

```

## B Presentace

Tato příloha obsahuje zdrojové kódy s komentáři, které se vztahují k přípravě prezentací pomocí programu  $\text{\LaTeX}$  ve třídě Beamer.

### B.1 Použití některých příkazů

#### B.1.1 Příkaz author

```

\author[autor] % bude zobrazeno v zahlaví, zapati slidu,
                % dle zvoleného tematu
% další parametr je pro vyplnění jména/jmen autora/autoru
% jsou tři možnosti, lze použít vždy jen jednu
% autor je jeden
  {autor}
% autoru je více (jsou ze stejné instituce)
  {autor1 \and autor2}
% autoru je více (jsou z různých institucí)
% pomocí příkazu \inst{} se jméno propojí s danou institucí
  {autor1\inst{1} \and autor2\inst{2}}
% lze použít jen jednu z nabízených možností

```

#### B.1.2 Příkaz institute

```

\institute[ČVUT] % instituce pro zapati - zkratka
% Další parametr lze zapsat dvěma možnostmi:
% pro jednoho autora nebo pro více autorů ze stejné
% organizace se vypíše název instituce
  {ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE\
  Katedra mapování a kartografie}
% pro více autorů, každý z jiné instituce se použije
% struktura
  {\inst{1}%
  Katedra1 \\ Univerzita1
  \and
  \inst{2}
  Katedra2 \\ Univerzita2}
% lze použít jen jednu z nabízených možností

```

### B.1.3 Příkaz `AtBeginSubsection`

Tento příkaz zajistí zobrazení daného slidu před každou sekci. V následující ukázce je daným slidem obsah, ve kterém se postupně zvýrazňují aktuální sekce, příp. podsekce. Tedy vždy před slidem, který popisuje problematiku nějaké sekce, se zobrazí obsah, kde je tato sekce zvýrazněna.

```

\AtBeginSubsection[]      % prikaz pro aktivaci zobrazeni
                          % nejakeho slidu pred kazdou subsekcí
{
  % obsah slidu
  \begin{frame}          % vytvoreni slidu
    \frametitle{Obsah}   % nazev slidu
    \tableofcontents     % vlozeni obsahu
      [currentsection,   % se zvýraznením aktuální sekce,
       currentsubsection] % subsekce
  \end{frame}           % konec slidu
}

```

## B.2 Jednoduchá prezentace

Zdrojový kód jednoduché prezentace může vypadat např. takto:

```

\documentclass{beamer}
\mode<presentation>
\usetheme{Berkeley}
\usepackage[czech]{babel}      % cestina pres babel: nutne prekladat
                                % pomoci pdflatex !!ne pdfcslatex!!
\usepackage[cp1250]{inputenc} % kodovani(cp1250 - vhodne pro Windows)
\title[DP]{Diplomová práce}
\subtitle{\LaTeX}
\author[J. Novák]{Jiří NOVÁK}
\institute[ČVUT]{ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE\
                Katedra mapování a kartografie}
\date[1. 1. 2009]
{1. ledna 2009}
\logo{\includegraphics[scale=0.5]{logo.pdf} }

```

```

\AtBeginSubsection[]
{\begin{frame}
  \frametitle{Obsah}
% obsah se zvýrazněním aktuální sekce, subsekce
  \tableofcontents[currentsection,currentsubsection]
\end{frame}
}
\begin{document}
\begin{frame}
  \titlepage
\end{frame}

\begin{frame}
  \frametitle{Obsah}
  \tableofcontents [pausesections] % dělá pauzy mezi jednotlivými
                                  % částmi (sekcemi) prezentace
                                  % v obsahu
\end{frame}

\section*{Cíle práce}
\begin{frame}
  \frametitle{Cíle práce}
  \begin{itemize}
    \item Seznámení s {\LaTeX}em
    \item Prozkoumání možností třídy Beamer
    \item Vytvoření prezentace
  \end{itemize}
\end{frame}

\section{\LaTeX}
\subsection[Historie]{Historie \LaTeX u}
\begin{frame}
  \frametitle{Historie \LaTeX u}
  \begin{itemize}
    \item Kdy vznikl?
    \item Kdo ho vymyslel?
    \item Proč byl vytvořen?
  \end{itemize}
\end{frame}

```

```

\subsection[Editory]{Editory vhodné pro \LaTeX}
\begin{frame}
\frametitle{Texmaker}
\begin{itemize}
\item Jednoduchý
\item Přehledný
\item Freeware
\end{itemize}
\end{frame}
\begin{frame}
\frametitle{LyX}
\begin{itemize}
\item Jednoduchý
\item Podobný MS Word
\item Freeware
\item Postup překladu
\end{itemize}
\end{frame}
\section{Prezentace}
\subsection{Beamer}
\begin{frame}
\frametitle{Beamer}
Beamer je třída vhodná pro tvorbu prezentací v \LaTeX u.
\end{frame}
\subsection[Postup]{Postup vytvoření prezentace}
\begin{frame}
\frametitle{Postup vytvoření prezentace}
\begin{enumerate}
\item Instalace Beameru -- Obdobná jako při přidávání balíků
\item Šablona -- Vhodná šablona usnadní práci
\item Překlad zdrojového souboru -- Musí být proveden
příkazem \texttt{pdflatex.exe}.
\item Otevření výsledné prezentace -- Použije se Acrobat
Reader, pro zvětšení prezentace na celý monitor
použijte klávesovou zkratku Ctrl+L.
\end{enumerate}
\end{frame}
\section*{Shrnutí}

```

```
\begin{frame}
\frametitle{Shrnutí I}
Zde je místo pro shrnutí celé prezentace.
\end{frame}

\begin{frame}
\frametitle{Shrnutí II}
Prezentace obsahuje mnoho informací, tak je shrnutí
delší a pokračuje i~na dalším slidu.
\end{frame}
\end{document}
```

## C Přiložené CD

Přiložené CD má následující strukturu a obsah:

- Diplomová práce
  - DP.pdf – diplomová práce
- HTML
  - htlatex – vzor a výsledek použití příkazu htlatex pokus.tex
    - \* logo.eps – vzorový obrázek
    - \* pokus.tex – jednoduchý vzorový soubor, který lze přeložit do html
    - \* pokus.html – výsledný soubor
    - \* další soubory, které vzniknou při konverzi
  - htlatex2 – vzor a výsledek použití příkazu htlatex pokus.tex "html,2"
    - \* logo.eps – vzorový obrázek
    - \* pokus.tex – jednoduchý vzorový soubor, který lze přeložit do html
    - \* pokus.html – hlavní výsledný soubor
    - \* pokussN.html – ostatní výsledné soubory, N je číslo sekce
    - \* další soubory, které vzniknou při konverzi
- Prezentace
  - Ukazky – ukázky prezentací
    - \* CVUT1.pdf
    - \* CVUT2.pdf
    - \* CVUT3.pdf
  - CVUT.tex – šablona pro psaní prezentací
  - logo.pdf – logo ČVUT
  - mapka.pdf – vzorový obrázek
- Šablona
  - obrazky – adresář, který obsahuje všechny obrázky studentské práce
  - text – text studentské práce
  - sablona.pdf – ukázka šablony
  - sablona.tex – šablona pro psaní studentských prací na ČVUT
  - thesiscvut.sty – styl pro psaní studentských prací na ČVUT