

Šimon Konečný, Petr Olšák

Pro zpracování \TeX em je tedy třeba mít odpovídající verzi základního textu (předzpracovanou způsobem, popsaným v sekci 2.4) a dále soubory s poznámkami (jejich syntaxe je uvedena v sekci 3). \TeX přečte hlavní soubor se základními pokyny, pak načte data všech potřebných souborů a vytvoří PDF soubor knihy – komentované Bible. V ní je jednak (typicky větším písmem) základní text a (typicky menším písmem, například pod čarou a třeba ve dvou sloupcích) poznámky k textu.

Je respektován základní požadavek, že k jednotlivým frázím základního textu jsou poznámky umístěny na stejné stránce jako samotné fráze základního textu. O rozmístění objektů na stránkách se \LaTeX stará víceméně automaticky podle nastavených typografických pravidel, které se dají

modifikovat dle potřeby. Požadavky na typografickou úpravu nebo další specifické požadavky, které se dají algoritmicky popsat, jsou implementovány makry \TeX u a \TeX je během zpracování přečte a automaticky provede.

Veškerá data, která \TeX zpracovává (konfigurace, základní text, texty poznámek, maker) mají textový formát, tj. dají se prohlížet a případně upravovat běžným textovým editorem.

Obsah

1	Spuštění \TeX u	3
2	Struktura souborů zpracovávaných \TeX em	3
2.1	Hlavní soubor s údaji o všech dalších souborech	3
2.2	Soubor deklarující varianty překladu	4
2.3	Soubor s názvy knih	5
2.4	Formát základního textu, tzv. .txs soubory	6
2.5	Údaje určující formátování základního textu v souborech <code>fmt-*.tex</code>	7
2.6	Poznámky v souborech <code>notes-*.tex</code>	7
3	Formát poznámek s vazbou na fráze v základním textu	7
3.1	Příkaz <code>\Note</code>	7
3.2	Formát příkazů pro vložení dalších objektů	8
3.3	Vložení obrázků	8
3.4	Vložení článků	9
3.5	Vložení citací do horní části stránky	9
3.6	Vložení citací do okraje v textu článku	10
4	Různé (ale podobné) verze základního textu	10
4.1	Deklarace variant překladu a použití pomoci příkazu <code>\x</code>	10
4.2	Variantní deklarace frází pro spárování poznámek s textem	11
4.3	Větvení zpracování textu podle variant překladu	12
4.4	Přečíslování veršů podle variant překladu	12
5	Metody vytváření hyperlinkových odkazů	13
5.1	Základní pravidlo s úplnými údaji	13
5.2	Specifikátor odkazu	13
5.3	Výjimka pro formát úplného údaje u některých knih	13
5.4	Neúplný údaj	14
5.5	Formát pro rozsah veršů a pro úsek ve verši	14
5.6	Zatažení údaje	15
5.7	Přečíslování odkazu	15
5.8	Redukce údajů	15
5.9	Nepodařené odkazy, tj. odkazy na neexistující místo	15
5.10	Trasování odkazů	15
5.11	Odkazy na kapitoly a na knihy	15
5.12	Odkazy na stránky	16
6	Varianty formátování stránek	16
7	Možnosti hledání chyb	16
8	Rejstřík	16

1 Spuštění T_EXu

Balíček OP-Bible předpokládá použití T_EXového engine LuaT_EX se sadou výchozích maker OpT_EX, viz <http://petr.olsak.net/optex>.

Máte-li nainstalovanou T_EXovou distribuci na svém počítači (například T_EXlive 2022) a máte-li v systému k dispozici příkazový řádek, pak T_EX spustíte pomocí:

```
optex hlavni.tex
```

kde `hlavni.tex` je název hlavního souboru (může se jmenovat i jinak). Příkaz `optex` spustí LuaT_EX s OpT_EXem. Výsledkem zpracování je soubor `hlavni.pdf` a zpráva o zpracování je v log souboru `hlavni.log`.

Je možné též využít nástroj [Overleaf](#) s T_EXovou distribucí provozovanou na webovém serveru. Můžete vyjít z výchozího projektu [OP-Bible na Overleaf](#) a zkopírovat si jej do svého projektu. Práce s T_EXovou distribucí přímo v počítači je ale výrazně rychlejší a pohodlnější.

2 Struktura souborů zpracovávaných T_EXem

2.1 Hlavní soubor s údaji o všech dalších souborech

Hlavní soubor je soubor, který se T_EXu předloží jako první. Například je uveden na příkazovém řádku pro spuštění T_EXu. V něm jsou informace, jaké další soubory si má T_EX přečíst. Nakonec T_EX vytvoří soubor PDF stejného názvu, jako je název hlavního souboru.

Hlavní soubor pro použití OP-Bible může vypadat takto:

```
\load[op-bible] % macros OP-Bible
\cslang

\def\tmark      {BKR}      % Variant of translation
\input {Cze-vars.tex}      % Variants o transtarion
\input {Cze-books.tex}     % Book titles and marks

\def\txsfile     {sources/Cze\tmark-\amark.txs}      % Localization of .txs files
\def\fmtfile     {formats/fmt-Cze\tmark-\amark.tex}  % Localisation of fmts
\def\notesfile   {notes/notes-\amark.tex}           % Localisation of notes
\def\introfile   {others/intro-\amark.tex}           % Localisation of introductions
\def\articlefile {others/articles-\amark.tex}        % Localisation of articles

\def\printedbooks {%
  Gn Ex Lv Nu Dt Joz Sd Rt 1S 2S 1Kr 2Kr 1CPa 2Pa Ezd Neh
  Est Jb Ž PŘ Kaz Pís Iz Jr Pl Ez Da Oz Jl Am Abd Jon Mi
  Na Abk Sf Ag Za Mal
  Mt Mk L J Sk Ř 1K 2K Ga Ef Fp Ko 1Te 2Te 1Tm 2Tm
  Tt Fm Žd Jk 1Pt 2Pt 1J 2J 3J Ju Zj
}
\processbooks % Generates books declared in \printedbooks
\bye
```

Pomocí `\load[op-bible]` načte T_EX makra balíčku OP-Bible.

Příkaz `\cslang` nastaví české vzory dělení slov, předpokládá se tedy český text. Přitom `cs` je ISO zkratka jazyka, můžete použít i jiné jazyky: `\enlang` pro angličtinu, `\delang` pro němčinu, `\eslang` pro španělštinu atd. Všechny tyto jazykové možnosti jsou uvedeny v dokumentaci k OpT_EXu.

Příkaz `\def\tmark {<značka>}` definuje makro `\tmark` jako značku použitého překladu. Značky všech dostupných překladů jsou uvedeny v souboru `Cze-vars.tex`. Jednu z nich je třeba vybrat jako označení zrovna zpracovávaného překladu. BKR je značka pro překlad Bible kralické.

`\input {Cze-vars.tex}` přečte konfiguraci o variantách překladu ze souboru `Cze-vars.tex`. Soubor je v ukázce nazván `Cze-vars.tex`, protože obsahuje informace o českých překladech. Podrobněji viz sekci 2.2.

`\input {Cze-books.tex}` přečte informace o značkách knih Bible a jsou jim přiřazeny názvy knih. Podrobněji viz sekci 2.3.

Makrem `\txsfile` (definovaným pomocí příkazu `\def`) je určeno umístění `.txs` souborů v adresářové struktuře. Pro každou knihu Bible musí být připraven jeden `.txs` soubor obsahující základní text pro příslušnou knihu. Názvy souborů se liší podle značky knihy a máme-li více překladů, obsahuje název souboru také značku překladu. V makru `\txsfile` lze použít `\tmark` jako značku překladu a `\amark` nebo `\bmark` jako značku knihy. O značkách knih se píše v sekci 2.3, o formátu `.txs` souborů se píše v sekci 2.4. V ukázce je vymezeno, že `.txs` soubory jsou umístěny v adresáři `sources/` a mají názvy `Cze⟨značka-překladu⟩-⟨značka-knihy⟩.txs`, například tedy `CzeBKR-Gn.tex`.

Makrem `\fmtfile` je vymezeno umístění souborů upřesňující formátování základního textu. Každá kniha Bible má svůj formátovací soubor. O těchto souborech se píše v sekci 2.5.

Makrem `\notesfile` je vymezeno umístění souborů s poznámkami. Každá kniha Bible má svůj soubor s poznámkami. Poznámky mají vymezen odkaz na místo v základním textu a úkolem \TeX u je vytvořit stránky se základním textem a poznámkami k nim. Podrobněji o způsobu psaní souborů s poznámkami se píše v sekci 3. Všimněte si, že soubory s poznámkami jsou společné všem překladům, tj. nejsou zde zvlášť soubory odlišené pomocí `\tmark`. Pravidla psaní poznámek totiž umožní obsáhnout vyjadřovací možnosti různých překladů společným zápisem, ovšem v jednom jazyku (například češtině). Chcete-li psát poznámky pro zcela jiný jazyk, je třeba založit nový projekt s jinými `.txs` soubory, jinými formátovacími a poznámkovými soubory.

Makrem `\introfile` se specifikují soubory, kde jsou napsány úvody k jednotlivým knihám. Ke každé knize je možné vytvořit samostatný soubor s úvodem. Makrem `\articlefile` určují názvy souborů kde jsou články k jednotlivým knihám. **TODO:** dopsat...

Makro `\printedbooks` obsahuje značky těch knih, které chcete \TeX em zpracovat. V ukázce je požadavek na zpracování celé Bible, tedy všech 66 knih. Pokud děláte třeba jen testovací tisky, můžete zpracovat jen některé biblické knihy a mít v hlavním souboru tedy alternativní definici, například `\def\printedbooks{Da}`. Stačí ji uvést za definici pro celou Bibli, protože pozdější definice stejného makra má přednost.

Příkaz `\processbooks` spustí zpracování všech knih specifikovaných v makru `\printedbooks`. Pro každou knihu \TeX přečte příslušný základní text z `.txs` souboru, formátuje jej pomocí údajů z příslušného formátovacího souboru a připojí k nim poznámky z příslušného poznámkového souboru.

Příkaz `\bye` ukončí činnost \TeX u.

2.2 Soubor deklarující varianty překladu

Pracujeme-li s jedinou variantou překladu, není nutné tento soubor vytvářet a používat. Pak stačí odstranit pokyn k jeho čtení z hlavního souboru.

V ukázce v sekci 2.1 je čten soubor `Cze-vars.tex`, který by měl obsahovat deklaraci značek variant překladů pomocí `\variants`:

```
\variants ⟨počet-variant⟩ {⟨značka⟩} {⟨značka⟩} ... {⟨značka⟩}
```

kde `⟨počet-variant⟩` je počet variant překladů (do společného jazyka, například češtiny) a pak jsou vyjmenovány všechny značky variant překladů. Například

```
\variants 6 {BKR} {PSP} {CSP} {CEP} {B21} {SNC}
```

deklaruje zkratky pro 6 variant překladů: BKR: Bible kralická, PSP: Pavlíkův studijní překlad, CSP: Český studijní překlad, CEP: Český ekumenický překlad, B21: Bible pro 21. století, SNC: Slovo na cestu.

Dále je možné v tomto souboru definovat variantní fráze jinak znějící v různých překladech pomocí příkazu `\vdef`. Frází uvedených za `\vdef` musí být přesně tolik, jako je `⟨počet-variant⟩`, jsou uzavřeny do složených závorek a korespondují s jednotlivými variantami překladu ve stejném pořadí, jak jsou varianty uvedeny v deklaraci `\variants`. Například:

```
\vdef {Joakim} {Jehójakim} {Jójakim} {Jójakim} {Joakim} {Jójakim}
```

deklaruje, že jméno Judského krále Joakima je přepisováno jinak v různých variantách překladu. Například v CEP nebo v SNC to je Jójakim, Pavlík jej nazývá Jehójakímem atd. Až budeme psát poznámky týkající se tohoto krále do poznámkového souboru, napíšeme do textu poznámky jen `\x/Joakim/` a toto se promění v odpovídající frázi používanou v právě zpracovávaném překladu, který jsme deklarovali v hlavním souboru pomocí `\def\tmark{...}`. Takže po změně `\def\tmark{...}` v hlavním souboru se všechny výskyty `\x/Joakim/` v textech poznámek mohou začít automaticky chovat jinak a přizpůsobí se frazeologii dané varianty překladu. Podrobněji o tom pojednává sekce 4.

Příkaz `\variants` deklarující zkratky variant překladu je v souboru pro varianty jediný, zatímco příkazů `\vdef` definujících variantní vráže může být v souboru více, protože pochopitelně existuje více frází, které se v různých variantách překladu různě přepisují, není to jen král Joakim.

2.3 Soubor s názvy knih

V hlavním souboru je pokyn na čtení souboru s názvy knih, například `\input {Cze-books.tex}`. Tento soubor musí obsahovat příkazy `\BookTitle` ve formátu:

```
\BookTitle <a-značka> <b-značka> {\titul knihy}
```

Mezi značkami a titulem knihy musí být aspoň jedna mezera. Začátek takto čteného souboru může vypadat třeba takto:

```
\BookTitle Gn    Gen    {První Mojžíšova (Genesis)}
\BookTitle Ex    Exod    {Druhá Mojžíšova (Exodus)}
\BookTitle Lv    Lev     {Třetí Mojžíšova (Leviticus)}
\BookTitle Nu    Num     {Čtvrtá Mojžíšova (Numeri)}
\BookTitle Dt    Deut    {Pátá Mojžíšova (Deuteronomium)}
\BookTitle Joz    Josh    {Jozue}
\BookTitle Sd    Judg    {Soudců}
...
```

V prvním sloupci za `\BookTitle` jsou *<a-značky>*, které jsou dále používány v textu poznámek a jsou pomocí nich tvořeny odkazy na různá místa v Bibli.

Ve druhém sloupci jsou *<b-značky>*, které mohou být stejné jako *<a-značky>*, ale mohou se také lišit. Je třeba možné, že názvy `.txs` souborů vznikly exportem z nějakého softwaru a značky knih jsou tam jiné, než potřebujeme použít v textu našich poznámek. Pak je možné v hlavním souboru deklarovat umístění `.txs` souborů s využitím `\bmark` místo `\amark`, tedy

```
\def\txsfile    {sources/Cze\bookmark-\bmark.tx}
```

a mít soubory `CzeBKR-Gen.tx`, `CzeBKR-Exod.tx`, zatímco v textu používáme značky `Gn`, `Ex`, atd.

Makro `\amark` obsahuje *<a-značku>* právě zpracovávané knihy a dále makro `\bmark` zahrnuje *<b-značku>* právě zpracovávané knihy.

Poznamenejme, že makro `\printedbooks` se značkami všech knih, které chceme zpracovat (viz sekci 2.1), obsahuje *<a-značky>*, nikoli *<b-značky>*.

Ve třetím parametru za `\BookTitle` jsou názvy knih ve složených závorkách.

Soubor s 66 údaji `\BookTitle` je vygenerován automaticky po extrakci základních textů z *Swordu* pomocí `mod2tex` a `maketxs` (viz sekci 2.4). Je možné jej použít, ovšem *<a-značky>* a tituly knih je potřeba asi manuálně upravit podle zvyklostí daného překladu, jak je předvedeno v ukázce výše.

Do souboru s názvy knih je možné přidat další informace o jednotlivých knihách pomocí příkazů `\BookException`, `\BookPre`, `\BookPost`. Mají tuto syntaxi:

```
\BookException <a-mark> {\text-výjimky}
\BookPre       <a-mark> {\text-před-knihou}
\BookPost      <a-mark> {\text-po-knize}
```

Přitom *<text-výjimky>* se vloží před čtením všech souborů knihy vymezené značkou *<a-mark>* uvnitř cyklu pro čtení všech knih příkazem `\processbooks`, dále *<text-před-knihou>* se vloží po načtení souborů knihy ale před zpracováním prvního verše. Konečně *<text-po-knize>* se vloží za posledním vešem knihy.

Příklad použití `\BookException`: pokud chcete mít v názvech souborů *<a-značky>* a ty obsahují diakritická znaménka, je možné upravit hodnotu makra `\amark` tak, aby neobsahovalo diakritiku takto:

```
\BookException Ž    {\def\amark{Z}}
\BookException Př    {\def\amark{Pr}}
\BookException Pís    {\def\amark{Pis}}
\BookException Ř     {\def\amark{R}}
\BookException Žd     {\def\amark{Zd}}
```

Pak třeba soubory `notes-\amark.tex` jsou ve skutečnosti pojmenovány `notes-Z.tex`, `notes-Pr.tex` atd. Přitom v textu poznámek dále normálně používáte značky knih `Ž`, `Př`, `Pís` atd.

Soubor s názvy knih by měl také obsahovat informaci o všech knihách, které nejsou členěny na kapitoly. Při odkazování na takové knihy se pak použije jiný způsob interpretace odkazu, viz sekci 5.3. To provedeme definováním makra `\nochapbooks`, které musí obsahovat *a-značky* těchto knih: `\def\nochapbooks {Abd Fm 2Jn 3Jn Ju}`.

2.4 Formát základního textu, tzv. .txs soubory

Předpokládá se, že základní text Bible je uložen v souborech s příponou .txs (text source). Každá z 66 knih Bible je uložena ve svém .txs souboru. Názvy .txs souborů a jejich umístění musejí odpovídat deklaraci `\def\txsfile` v hlavním souboru (viz sekci 2.1).

Každý řádek .txs souboru obsahuje jeden verš Bible uvozený `#<číslo-kapitoly>:<číslo-verše>`. Verše musejí být uvedeny ve správném pořadí. Například začátek souboru CzeBKR-Da.txs vypadá takto (části textu jsou v ukázce vynechány):

```
#1:1 Léta třetího kralování Joakima krále Judského, ... a oblehl jej.
#1:2 I vydal Pán v ruku jeho Joakima ... do domu pokladu boha svého.
...
```

Základní texty Bible lze získat například z modulů Swordu <https://www.crosswire.org/sword/modules/ModDisp.jsp?modType=Bibles>. Jednotlivé .txs soubory je pak možné vygenerovat následujícím postupem.

ZIP stažený z uvedené www stránky (tzv modul) někam rozbalte. Je třeba mít dále v počítači instalován balíček libsword-dev a dále program mod2tex, který je součástí OP-Bible. Pomocí `installmgr -l` zjistíte seznam stažených modulů. Máte-li nastaven aktuální adresář v místě, kam jste rozbalili ZIPy, a vznikl tam adresář `modules`, pak jsou takové moduly nalezeny. Moduly obsahují texty v binárním formátu, my je potřebujeme převést do textového formátu. K tomu stačí napsat do příkazového řádku:

```
mod2tex modul > soubor
```

kde `modul` je název modulu. Ve výsledném souboru máte kompletní základní text daného překladu (modulu). Například po

```
mod2tex CzeBKR > CzeBKR.out
```

je v souboru `CzeBKR.out` kompletní překlad Bible kralické. Ten nyní můžete rozdělit do .txs souborů příkazem

```
maketxs CzeBKR.out
```

Uvedený příkaz kromě 66 .txs souborů vytvoří soubor `CzeBKR-books.tex`, ve kterém jsou tituly a zkratky jednotlivých knih, tedy je tam:

```
\BookTitle Gen Gen {Genesis}
\BookTitle Exod Exod {Exodus}
\BookTitle Lev Lev {Leviticus}
\BookTitle Num Num {Numbers}
\BookTitle Deut Deut {Deuteronomy}
\BookTitle Josh Josh {Joshua}
\BookTitle Judg Judg {Judges}
...
```

Tyto tituly nejsou v češtině, protože je zdroj Sword neobsahuje. Je tedy třeba tento soubor manuálně upravit a vložit místo anglických české názvy. Zkratky knih jsou uvedeny dvakrát stejně. První pozměňte podle zvyklostí používání zkratk v českých Biblích a druhou zkratku můžete ponechat. Pak deklarace

```
\def\txsfile {sources/Cze\tmark-\bmark.txs}
```

v hlavním souboru způsobí, že se vytvořené .txs soubory budou hledat v adresáři `sources/` a předpokládají se jejich názvy `CzeBKR-Gen.tx`s, `CzeBKR-Exod.tx`s, `CzeBKR-Lev.tx`s, atd.

Se soubory nadále nebudete potřebovat nijak manipulovat, není potřebné je editovat ani doplňovat. V adresáři `sources/` tedy můžete mít „datový sklad“ všech základních textů Bible pro všechny použité varianty překlady najednou. V případě šesti variant překladu tam tedy máte $6 \times 66 = 396$ souborů.

2.5 Údaje určující formátování základního textu v souborech `fmt-*.tex`

Základní text v `.tex` souborech neobsahuje žádné formátovací ani doplňkové údaje, jakými jsou například názvy kapitol nebo místa, kde se má ukončit odstavec či přejít z formátování v bloku do formátování na střed řádků a zpět.

Potože do základního textu nechceme nijak zasahovat, je třeba tyto doplňkové údaje deklarovat k příslušným veršům pomocí speciálních příkazů `\fmtadd`, `\fmtpre` a `\fmtins`. Tyto příkazy jsou typicky v souborech `fmt-*.tex`, například `fmt-CzeBKR-Da.tex`. Je vhodné udržovat tyto formátovací soubory závislé jednak na knize (Daniel v uvedeném příkladu), ale také na použitém překladu (Bible kralická v příkladu). Je sice možné začít s jedním souborem pro každou knihu a soubory pro ostatní překlady pořídit jako kopie výchozích, ale nakonec bude asi potřebné pro rozličné varianty základního textu formátovací pokyny mírně modifikovat v souladu s použitým překladem.

Syntaxe použití uvedených příkazů je následující:

```
\fmtpre{<číslo-kapitoly>: <číslo-verše>}{<fmt-pokyn>}
\fmtadd{<číslo-kapitoly>: <číslo-verše>}{<fmt-pokyn>}
\fmtins{<číslo-kapitoly>: <číslo-verše>}{<fráze>}{<fmt-pokyn>}
```

kde `<fmt-pokyn>` je „vzkaz“, který je předán \TeX u k formátování. Například `\endgraf` značí konec odstavce. `\begcenter` otvírá pasáž s centrováním textem a ta musí být uzavřena pomocí `\endcenter`. Nebo `\chaptit{<text>}` vloží `<text>` jako titul kapitoly.

Příkaz `\fmtpre` vkládá `<fmt-pokyn>` na začátek uvedeného verše (ještě před případně vytištěné číslo verše horním indexem). Příkaz `\fmtadd` vkládá `<fmt-pokyn>` na konec uvedeného verše. Konečně `\fmtins` vkládá `<fmt-pokyn>` dovnitř verše za první výskyt stanovené `<fráze>`, která ve verši musí doslovně existovat. Jinak \TeX vypíše varování a `<fmt-pokyn>` nezařadí vůbec.

Jak může vypadat použití příkazů `\fmt*` je vidět například v souboru `fmt-BKR-Da.tex`.

2.6 Poznámky v souborech `notes-*.tex`

Tyto soubory (umístěné podle deklarace `\notesfile` v hlavním souboru) obsahují poznámky k jednotlivým veršům nebo částem veršů. K psaní poznámek slouží zejména příkaz `\Note`. Protože toto je hlavní důvod, proč byl vytvořen balíček OP-Bible, je tomu věnována celá další kapitola 3.

3 Formát poznámek s vazbou na fráze v základním textu

3.1 Příkaz `\Note`

Hlavním smyslem nástroje OP-Bible je vytvořit PDF ze základního textu nejen se samotným textem Bible, ale s propojenými poznámkami. K tomu slouží mimo jiné soubory `notes-*.tex` (například `notes-Gn.tex` pro knihu Genesis), ve kterých jsou zapsány poznámky k jednotlivým frázím základního textu dle následujícího smlouveného formátu. Jednotlivá poznámka je uvozena příkazem `\Note` a má tvar:

```
\Note <číslo-kapitoly>: <číslo-verše> {<fráze>} <text-poznámky>
<prázdný řádek>
```

Například:

```
\Note 1:2 {nádobí} Odkaz na nádobí z vypleněného chrámu, nikoliv na deportaci zajatců.
```

Příklad je ze souboru `notes-Da.tex`, tedy ze souboru s poznámkami ke knize Daniel. Konkrétně se poznámka váže na kapitolu první, verš druhý a na frázi „nádobí“. Tato fráze musí být bezpodmínečně ve stanoveném verši základního textu obsažena. Pak ji \TeX propojí s odpovídajícím místem v základním textu, tj. zajistí, že výskyt fráze i samotné poznámky je na společné stránce.

Povšimněme si, že fráze se váže na překlad Bible kralické, zatímco v překladu ekumenickém je použit ve stejném verši termín „nádob“. Jak zařídit, aby se stejná poznámka automaticky mírně modifikovala podle použitého překladu, je popsáno v sekci 4.2.

Pokud `<fráze>` není ve stanoveném verši základního textu doslovně obsažena, \TeX během zpracování oznámí do logu a a na terminál o tom varování a poznámku zařadí k danému verši, jakoby `<fráze>` byla na začátku verše..

Vytištěná poznámka (v závislosti na typografickém nastavení) obsahuje například zopakování čísla kapitoly a verše, pak následuje komentovaná fráze (například tučně) a za ní je vlastní text poznámky.

Někdy je třeba v základním textu vyhledat mírně jinou frázi, než jakou pak chceme zapsat do vytištěné poznámky (např. v základním textu je slovo v jiném pádě nebo je to mírně jinak formulovaná fráze). Pak je možné těsně za `{(fráze)}` napsat rovnítko následované `{(fráze-k-tisku)}`, tedy

```
\Note <číslo-kapitoly>:<číslo-verše> {(fráze)}={{(fráze k tisku)}} <text-poznámky>
<prázdný řádek>
```

V takovém případě se v základním textu hledá `<fráze>`, ale ve vlastní poznámce se vytiskne jako heslo poznámky `<fráze-k-tisku>`. Například:

```
\Note 1:2 {vydal Pán}={Pán vydal} Porážka Izraele Babylónem není vysvětlitelná
jen pouhou vojenskou a politickou analýzou oné doby. Bůh vždy jednal svrchovaně
v záležitostech národů. Babylóňany použil jako nástroj potrestání svého vlastního
lidu za porušení smluvních závazků.
```

V tomto příkladě se hledá ve verši druhém základního textu „vydal Pán“ ale v poznámce je vytištěno heslo „Pán vydal“.

Povšimněte si, že tato poznámka se rovněž váže na verš druhý kapitoly první knihy Daniel (protože je v souboru `notes-Da.tex`). Je tedy možné mít více poznámek k různým frázím, které se váží na stejný biblický verš.

Jednotlivé poznámky ve zdrojovém souboru poznámek jsou odděleny prázdnými řádky. To je nutné, jinak by \TeX při jejich čtení nepoznal, kde končí text poznámky. Také se tím zvyšuje přehlednost zdrojového souboru. Další řádky poznámky můžete (ale nemusíte) odsadit. Obecná pravidla pro zápis zdrojových textů pro \TeX nejsou předmětem tohoto manuálu, najdete je v různých učebnicích k \TeX u.

Pokud se poznámka váže k celému verši (tj. bez specifikované fráze), pište `{}`, tedy prázdnou frází k vyhledání. Například:

```
\Note 1:1-21 {}={Uchování rituální čistoty} Prorok uvozuje kontext své knihy vyprávěním
osobní historie (své i svých přátel) zajetí, vzdělání, věrnosti Bohu a služby
králi Nabuchodonozorovi.
```

Tato ukázka navíc demonstruje, že je také možné uvést celý rozsah čísel veršů. Je-li `<fráze>` pro vyhledání prázdná (jako v tomto příkladě), pak se poznámka umístí na stejnou stranu, kde je začátek prvního verše z uvedeného rozsahu. Je-li `<fráze>` neprázdná, musí se vyskytovat v prvním verši z uvedeného rozsahu veršů. Rozsah veršů bude stejně vytištěn v úvodu poznámky. Symbol pro rozsah „-“ je jediný znak „mínus“ běžně dosažitelný na klávesnici. Nesmí to být žádný speciální znak vypadající třeba podobně jako vodorovná čárka.

Pořadí vytištěných poznámek k jednomu verši odpovídá pořadí frází, na které se váží, v základním textu. Na pořadí, ve kterém jsou zapsány ve zdrojovém souboru to tedy nemá vliv. Poznámky, které se váží pomocí `{}` k celému verši jsou řazeny jako první a je-li jich více, pak jsou řazeny podle pořadí ve zdrojovém souboru.

3.2 Formát příkazů pro vložení dalších objektů

Do souboru `notes-*.tex` je možné kromě příkazů `\Note` zapisovat pokyny pro vložení dalších objektů s vazbou na konkrétní verš v základním textu. Takový objekt se umístí¹ na spodní část strany pod dva sloupce poznámek nebo citáty se umístí na horní část strany. Vazbu na místo textu vymezíme pomocí údaje `<číslo-kapitoly>:<číslo-verše>` podobně, jako v případě poznámek `\Note`. Objekt se umístí na stejnou stranu, kde začíná příslušný verš. Pokud se na stránku pod verš objekt nevejde, umístí se na spodní část následující stránky.

3.3 Vložení obrázků

Obrázky s vazbou na konkrétní verš umístíme příkazem `\putImage`:

```
\putImage <číslo-kapitoly>:<číslo-verše> {<titulek>} [<lejblík>] (<parametry>) {<soubor>}
```

¹ Podle implicitní definice vzhledu jednotlivých stran, to je možné předefinovat.

Například:

```
\putImage 2:1 {Danielovy vize} [danielovyvize] () {Nabuco.pdf}
```

- $\langle\text{číslo-kapitoly}\rangle:\langle\text{číslo-verše}\rangle$ označuje místo, ke kterému se obrázek umístí.
- $\langle\text{titulek}\rangle$ je titulek (popisek) k obrázku. V implicitním typografickém nastavení se vytiskne do proužku, který odděluje horní část strany od samotného obrázku.
- $\langle\text{lejblík}\rangle$ udává lejblík k obrázku pro případné reference. Můžete pak na něj odkazovat pomocí ...viz obrázek $\backslash\text{ref}[\langle\text{lejblík}\rangle]$ na straně $\backslash\text{pg}$, což vytiskne v případě naší ukázky ...viz obrázek [Danielovy vize](#) na straně 123. Přitom text v tomto odkazu bude propojen hyperlinkem s místem, kde se vyskytuje obrázek. Když nechcete $\langle\text{lejblík}\rangle$ použít, pište $[\]$ (tj. jakoby prázdný lejblík).
- $\langle\text{parametry}\rangle$ určují případný další pokyn pro formátování obrázku. Mohou chybět, jak naznačuje ukázka. Implicitně se obrázek roztáhne na celou šířku stránky. Pokud ho chete mít třeba menší, pište do místa pro $\langle\text{parametry}\rangle$ třeba $\backslash\text{picw}=9\text{cm}$ což způsobí, že obrázek bude mít šířku 9 cm a bude umístěn na střed.
- $\langle\text{soubor}\rangle$ je úplné jméno souboru s obrázkem. Může mít příponu pdf (pro vektorové obrázky) nebo png či jpg pro bitmapové obrázky. Soubory s obrázky musejí být uloženy v adresáři images/. Jsou-li umístěny jinde, je třeba nastavit pro toto umístění parametr $\backslash\text{picdir}$, například $\backslash\text{picdir}=\{\text{bibleImages}/\}$. Toto je rozumné nastavit v hlavním souboru, viz sekci 2.1.

3.4 Vložení článků

Článek s textem (typicky delším než poznámka) lze umístit podobně jako obrázky, tj. při implicitním typografickém nastavení na spodní stranu stránky, na níž začíná verš specifikovaný pomocí $\langle\text{číslo-kapitoly}\rangle:\langle\text{číslo-verše}\rangle$. Nevejde-li se tam, vloží se na spodní část následující strany. Nevejde-li se na jedinou stranu, vloží se jeho další úsek na spodní část následující strany (atd. až do chvíle, kdy je celý článek do textu vložen).

Text článků ke každé knize Bible zapisujte do souboru nazvaného a uloženého podle makra $\backslash\text{articlefile}$, například v souboru articles-Gn.tex jsou všechny články pro knihu Genesis. Jak obsah tohoto souboru vypadá popíšeme za chvíli.

V souboru notes-*.tex je třeba uvést požadavek na umístění článku pomocí $\backslash\text{putArticle}$:

```
\putArticle <číslo-kapitoly>:<číslo-verše> {\titulek} [<číslo-článku>] (<parametry>)
```

Například:

```
\putArticle 6:1 {Kdo byl Darius Médský?} [6] ()
```

- $\langle\text{číslo-kapitoly}\rangle:\langle\text{číslo-verše}\rangle$ označuje místo, ke kterému se obrázek umístí.
- $\langle\text{titulek}\rangle$ je titulek ke článku. Vytiskne se podobně jako v případě obrázku.
- $\langle\text{číslo-článku}\rangle$ je nějaké číselné označení článku, které musí být v rámci jedné knihy jednoznačné. Typicky se články váží na kapitoly, pak by $\langle\text{číslo-článku}\rangle$ mělo znamenat číslo odpovídající kapitoly. Podle tohoto čísla se článek do sazby neumísťuje (k tomu slouží údaj $\langle\text{číslo-kapitoly}\rangle:\langle\text{číslo-verše}\rangle$), nicméně je toto číslo použité v odkazech. Můžete například psát ...viz <"článek" Da 6>a a vytiskne se ...viz [článek Da 6](#). Formát a vlastnosti takových odkazů popisuje sekce 5. Navíc se podle $\langle\text{číslo-článku}\rangle$ vyhledá vlastní text článku v příslušném souboru articles-*.tex.
- $\langle\text{parametry}\rangle$ jsou parametry přesněji určující formátování článku. **TODO...**

Text článků ke vložení musí být uveden v souboru articles- $\langle\text{kniha}\rangle$.tex. Tento subor musí obsahovat řádek uvozený $\backslash\text{Article}$ takto:

```
\Article [<číslo-článku>]
```

Za ním následuje text článku. Pak může pokračovat další $\backslash\text{Article} [<číslo-článku>]$ následovaný další textem atd. V jednom souboru jsou tak shomážděny všechny články k dané knize.

Pokud zadáte požadavek $\backslash\text{putArticle}$ a příslušný soubor article- $\langle\text{kniha}\rangle$.tex nebo příslušný řádek v něm $\backslash\text{Article} [<číslo-článku>]$ neexistuje, překlad skončí s chybou.

3.5 Vložení citací do horní části stránky

Příkaz $\backslash\text{putCite} <číslo-kapitoly>:\langle\text{číslo-verše}\rangle \{\text{<text>}\}$ vloží při implicitním typografickém nastavení $\langle\text{text}\rangle$ ve formě citátu na začátek stránky, na které se nachází $\langle\text{číslo-kapitoly}\rangle:\langle\text{číslo-verše}\rangle$.

Samotný `<text>` může obsahovat na konci `\quotedby {<autor>}`. V takovém případě je vytištěn `<autor>` na novém řádku (při citátu umístěném na levé stránce) nebo za předpokladu, že je dost místa, je na posledním řádku posunutý více vpravo (při citátu umístěném na pravé stránce).

Pro řízený přechod na nový řádek můžete v `<textu>` použít příkaz `\nl`.

3.6 Vložení citací do okraje v textu článku

Text článků je v implicitním nastavení sazby formátován do dvou sloupců. Vnější sloupec je možné přerušit a do vzniklého místa vsunout např. citát či jinou myšlenku. Tento text bude vyčuhovat do vnějšího okraje.

Uvnitř článku (tedy těsně po příkazu `\Article` [`<číslo-článku>`]) je třeba vložit deklaraci citovaného textu pomocí: `\Cite <písmeno> {<text>}`. Zde `<písmeno>` je typicky A. Pokud ale chcete do jednoho článku vložit více citátů, je třeba je odlišit dalšími písmeny, tedy B, C, atd. a všechny citáty napište na začátek článku.

Příkazem `\Cite` je citát pouze deklarován. Jeho skutečné vložení do textu článku provedete pomocí `\insertCite <písmeno>\left` a současně na jinde použijte `\insertCite <písmeno>\right`. Je-li článek na levé stránce dvoustrany, citát se umístí jen podle `\insertCite <písmeno>\left`, jinými slovy `\insertCite <písmeno>\right` je ignorováno. Je-li článek na pravé stránce otevřené dvoustrany, umístění se řídí jen podle `\insertCite <písmeno>\right`. Hned pod řádkem, ve kterém je příslušné `\insertCite` uvedeno, se citát skutečně objeví. Řádek samotný se kvůli tomu nerozdělí (jinak řečeno, odstavec se kvůli vložení citátu neukončí).

Důvod, proč je nutné uvést dvě lokace `\insertCite` je následující. Dopředu nevíme, na které stránce (liché nebo sudé) se článek s vloženým citátem objeví. Protože citát má zasahovat do vnějšího okraje, je třeba jej na levé stránce umístit do prvního sloupce, na pravé stránce do druhého sloupce, tedy někde poněkud jinak. Lokace `\insertCite <písmeno>\left` by tedy měla odpovídat nějakému řádku v prvním sloupci a `\insertCite <písmeno>\right` jinému řádku v druhém sloupci článku.

Je rozumné si odladit (tj. dopředu vyzkoušet), jak vypadá umístění citátu pro obě varianty, které mohou nastat (vlevo/vpravo). Pokud se chcete podívat, jak to dopadne při variantě, která neodpovídá správné poloze stránky, můžete použít na začátku textu článku příkaz `\swapCites`. Toto by ale nemělo zůstat zapnuté pro definitivní tisk, proto aktivované `\swapCites` způsobí varování na terminál a do log souboru.

4 Různé (ale podobné) verze základního textu

Existuje například více českých překladů Bible, které používají různé zvyklosti pro vyjádření některých frází. Máme tedy různé varianty základního textu a chceme k nim připojit společný poznámkový aparát, ve kterém se budou označené fráze automaticky měnit podle zvoleného překladu. K tomu slouží příkazy popsané v této sekci.

4.1 Deklarace variant překladu a použití pomocí příkazu `\x`

Nejsou-li varianty deklarovány příkazem `\vdef`, pak příkaz `\x/<fráze>/` použitý v textu vypíše `<fráze>`. Je ovšem možné deklarovat varianty překladu, jak je uvedeno v sekci 2.2.

Pokud je nyní v hlavním souboru uvedeno: `\def\tmark {<varianta>}`, pak `\x/<fráze>/` se promění na `<frázi>` specifikované `<varianty>`. Přitom parametr `<fráze>` příkazu `\x` musí být shodný s první frází uvedenou v příkazu `\vdef`.

Ilustrujme si toto na příkladu. Předpokládejme, že máme soubor `vars.tex`, který je čten z hlavního souboru pomocí `\input` a soubor obsahuje deklaraci tří variant:

```
\variants 3 {BKR}           {CEP}           {CSP}
\vdef      {Baltazar}       {Beltšasar}     {Beltšasar}
\vdef      {Nabuchodonozor} {Nebúkadnesar} {Nebúkadnesar}
\vdef      {sedm let}       {sedm let}      {sedm časů}
```

Je-li v hlavním souboru `\def\tmark{BKR}`, pak

```
\x/Baltazar/      vypíše Baltazar
\x/Nabuchodonozor/ vypíše Nabuchodonozor
\x/sedm let/       vypíše sedm let
\x/cokoli/         vypíše cokoli
```

Je-li ale v hlavním souboru `\def\tmark{CEP}`, pak

```
\x/Baltazar/      vypíše Beltšasar
\x/Nabuchodonozor/ vypíše Nebúkadnesar
\x/sedm let/      vypíše sedm let
\x/cokoli/        vypíše cokoli a na terminálu bude varování,
                  že fráze /cokoli/ nemá deklarován překlad.
```

Například poznámka v knize Daniel uvedená v sekci 3 by mohla znít:

```
\Note 1:1-21 {}={Uchování rituální čistoty} Prorok uvozuje kontext své knihy vyprávěním
osobní historie (své i svých přátel) zajetí, vzdělání, věrnosti Bohu a služby
králi \x/Nabuchodonozor/ovi.
```

a tato poznámka při `\def\tmark{CEP}` na svém konci zní „... a služby králi Nebúkadnesarovi“, zatímco při `\def\tmark{BKR}` zní „... a služby králi Nabuchodonozorovi“.

Některé parametry příkazu `\vdef` mohou být prázdné (zapsané jako `{}`), což značí nedefinovanou frázi pro daný jazyk. Bude-li potřeba takovou frázi použít pomocí `\x/. . . /`, vypíše se varování. Dále parametr může obsahovat jediný znak „“, což znamená, že se použije stejná fráze, jako v předchozím parametru. Takže naše výše uvedená ukázka může také vypadat takto:

```
\variants 3 {BKR}      {CEP}      {CSP}
\vdef      {Baltazar}   {Beltšasar} {}
\vdef      {Nabuchodonozor} {Nebúkadnesar} {}
\vdef      {sedm let}    {}         {sedm časů}
```

4.2 Variantní deklarace frází pro spárování poznámek s textem

Před příkazem `\Note` může bezprostředně předcházet deklarace vyhledávaného slova podle variant překladu pomocí `\ww` (to je zkratka za watchword). Příkaz `\ww` má tolik parametrů, kolik je variant překladu deklarovaných příkazem `\variants` a tyto parametry mohou být jednoduché (ve formátu `{<hledaná-fráze>}`) nebo složené (ve formátu `{<hledaná-fráze>}{<co-vytisknout>}`). Bezprostředně následující `\Note` pak bude ignorovat svůj parametr pro hledanou frázi a použije parametr z `\ww` odpovídající zpracovávané variantě jazyka. Například:

```
\ww {liternímu umění a jazyku} %BKR
    {písmu a jazyku Kasdím} %PSP
    {chaldejskému písemnictví a jazyku} %CSP
    {kaldejskému písemnictví a jazyku} %CEP
    {babylonskému písemnictví a jazyku} %B21
    {kaldejský jazyk}={kaldejský jazyk ... s národním písemnictvím} %SNC
\Note 1:4 {} Babylónská literatura byla ...
```

Zde uvedená poznámka `Note 1:4` vyhledá ve verši 1:4 text „liternímu umění a jazyku“, když je zpracovávána varianta překladu BKR a vyhledá text „kaldejskému písemnictví a jazyku“ při zpracování varianty překladu CEP. Ukázka předpokládá, že bylo pomocí příkazu `\variants` deklarováno šest variant překladu ve stanoveném pořadí. Můžete též stanovit rozdílnou frázi pro vyhledání a pro vytištění v poznámce, jak ukazuje následující příklad:

```
\ww {Baltazar} %BKR
    {Béltešaccar}={Béltešaccar ... Šadrach ... Měšach ... i Avéd-negó} %PSP
    {Beltšasar}={Beltšasar ... Šadrak ... Měšak ... i Abed-nego} %CSP
    {Beltšasar}={Beltšasar, Chananjášovi Šadrak, Mišaelovi Měšak a Azarjášovi Abed-nego}
    {Baltazar}={Baltazar, Chananjášovi Šadrach, Mišaelovi Měšach a Azariášovi Abednego}
    {Beltšasarem}={Beltšasarem, Chananjáše Šadrakem, Mišaele Měšakem a Azarjáše Abed-negem}
\Note 1:7 {} Přesný význam těchto jmen je předmětem diskusí...
```

Fráze pro hledání a nahrazení se použijí přesně tak, jak jsou napsány v parametrech příkazu `\ww`. To neplatí pro poznámky `\Note`, které nemají před sebou `\ww`. Pak při použití

```
\Note <verš> {<hledaná-fráze>} <text> <prázdný řádek>
```

nebo

```
\Note <verš> {<hledaná-fráze>}{<co-vytisknout>} <text> <prázdný řádek>
```

se *<hladaná-fráze>* nejprve transformuje podle dat z `\vdef`. Teprve pokud tato data pro hledanou frázi neexistují, použije se *<hladaná-fráze>* tak, jak je.

4.3 Větvení zpracování textu podle variant překladu

Pomocí příkazu `\switch` můžete větvit zpracování vstupního textu v závislosti na nastavené hodnotě parametru `\tmark`, tedy v závislosti na aktuálně zpracovávané variantě jazyka. Příkaz má tuto syntaxi:

```
\switch {<seznam variant>} {<co provést>}%
      {<seznam variant>} {<co provést>}% ... atd
      {<seznam variant>} {<co provést>}
```

Dvojic `{<seznam variant>}{<co provést>}` může být uvedeno libovolně mnoho. Za každou dvojici `{<seznam variant>}{<co provést>}` (s výjimkou poslední dvojice) musí bezprostředně a bez mezer následovat další taková dvojice, proto je nutné při přechodu na další řádek napsat za uzavírací závorku procento, které zakryje mezeru z konce řádku. Mezery na začátku dalšího řádku nevadí. Proceto za dvojicí můžete číst jako „pokračuje další dvojice“.

<seznam variant> je jedna překladová varianta nebo seznam překladových variant oddělených čárkou a bez mezer. \TeX pak pracuje takto: vyskytuje-li se v *<seznam variant>* varianta definovaná v parametru `\tmark`, provede se následující *<co provést>*. Nevyskytuje-li se tam taková varianta, následující *<co provést>* se přeskóčí. Příklad:

```
\switch {BKR} {Balsazar}%
      {PSP} {Bélšaccar}%
      {CSP,CEP,B21,SNC} {Belšasar}
```

Příklad ukazuje, jak vytisknout slovo Balsazar v závislosti na zpracovávané variantě jazyka.

Jakmile \TeX najde shodu a provede *<co provést>*, pak případně následující údaje v rámci stejného příkazu `\switch` přeskóčí. Dále platí pravidlo, že pokud je *<seznam variant>* prázdný, provede se *<co provést>* vždy, pokud tedy není přeskóčeno dle předchozího pravidla. Takže prázdný *<seznam variant>* na konci dvojic parametrů příkazu `\switch` se vyhodnotí jako „ostatní případy“. Příklad uvedený výše se tedy dá zapsat také takto:

```
\switch {BKR} {Balsazar}%
      {PSP} {Bélšaccar}%
      {} {Belšasar}
```

Příkaz `\switch` se dá použít nejen na jednotlivé fráze uvnitř poznámek `\Note`, ale třeba také na celé úseky vstupního textu obsahující třeba i několik poznámek `\Note`, několik definic atd.

Příkaz `\switch` se nedá použít v parametrech jiných maker. Tam funguje jen příkaz `\x/{fráze}/`.

4.4 Přechíslování veršů podle variant překladu

Některé varianty překladu mají jiné číslování veršů. V takovém případě je možné použít příkaz `\renum` takto:

```
\renum <zkratka-knihy> <výchozí-č-kap>:<výchozí-č-verše> = <překlad> <č-kap>:<č-od>-<č-do>
```

kde *<překlad>* je značka pro konkrétní překlad. Místo *<výchozí-č-kap>:<výchozí-č-verše>* se použije při `\def\tmark{<překlad>}` ve skutečnosti *<č-kap>:<č-od>*. Takové přechíslování se netýká jen tohoto verše, ale celého úseku veršů vymezeného rozsahem *<č-od>-<č-do>*.

Například:

```
\renum Da 4:16 = CEP 4:13-34
```

znamená, že `\Note 4:16 {sedm let}` Sedm období neurčené ... je poznámka, která ve výchozím překladu *<varianta-A>* (BKR v našem předchozím příkladě) se váže na text *sedm let*, který je umístěn skutečně ve verši Da 4:16. Když ale je `\def\tmark{CEP}`, pak uvedená poznámka `\Note 4:16` se váže na verš Da 4:13 tohoto překladu, ve kterém je vyhledána fráze *sedm let*. A k přechíslování dojde i pro následující verše, takže 4:16 dle BKR je 4:13 dle CEP, 4:17 dle BKR je 4:14 dle CEP, atd., až 4:37 dle BKR je 4:34 dle CEP.

Má-li se přechíslování týkat jediného verše, je třeba uvést shodné *<č-od>* i *<č-do>*, tedy například

```
\renum Da 4:16 = CEP 4:13-13
```

5 Metody vytváření hyperlinkových odkazů

Odkaz je úsek textu, podle kterého čtenář i po vtištění pozná, na jaké jiné místo v textu (nebo v internetu) se může podívat. Obsahuje tedy typicky číselný údaj o stránce nebo o čísle kapitoly, sekce atd. Navíc, pokud čtenář pracuje s PDF prohlížečem, pak tento úsek textu je *aktivní*, tj. po najetí myším kurzorem nad tento text je možné kliknout a PDF prohlížeč přejde na stanovené místo v dokumentu (nebo do internetu). Bible je neměnným způsobem strukturovaný text. Obsahuje 66 knih s ustálenými značkami pro tyto knihy, každá kniha má své kapitoly číslované od jedné a každá kapitola má verše číslované od jedné. Není tedy potřeba nechat \TeX , aby generoval tato čísla automaticky (jako to dělá při sazbě třeba odborného textu členícího se na kapitoly a sekce) a není tedy vesměs potřeba používat lejbliky ve zdrojovém dokumentu (které \TeX při zpracování přiřadí k vygenerovaným číslům) a odkazovat pomocí těchto lejbliků, jak je popsáno v sekci 1.4.3 v dokumentaci Op \TeX u. Je daleko účelnější odkazovat přímo do konkrétního místa v Bibli, které má už po mnohá staletí pevné označení knihy, čísla kapitoly a čísla verše. A mnozí a jiní lidé tyto odkazy už dávno označovali ustáleným způsobem zkratkami knih a číslu.

Odkazy píšeme na konkrétní místo v Bibli mezi `<` a `>`. Text mezi těmito znaky se vytiskne tak, jak je napsán (až na odstranění znaků `"`). \TeX ale musí umět odkaz správně interpretovat, aby jej mohl udělat aktivní s možným proklikem na správné místo v Bibli. K tomu slouží poměrně značné množství pravidel popsaných níže.

5.1 Základní pravidlo s úplnými údaji

Odkaz mezi znaky `<` a `>` má tvar `"<text>" <údaj>` nebo jen `<údaj>`. Přitom úplný `<údaj>` má tvar `<kniha> <kapitola>: <verš>`. Zde `<kniha>` je zkratka knihy (za ní musí být mezera), `<kapitola>` je číslo kapitoly a `<verš>` je číslo verše. Příklad:

```
... viz také verš <Jr 8:13>
... viz také <"verš" Jr 8:13>
```

V prvním případě se vytiskne ... viz také verš [Jr 8:13](#) a ve druhém ... viz také [verš Jr 8:13](#). Aktivní (umožňující proklik) bude v obou případech oblast zde vyznačena modře a proklik vede na Jr 8:13.

5.2 Specifikátor odkazu

Za koncovým znakem odkazu `>` může těsně následovat specifikátor odkazu, což je jedno z písmen:

- `n` ... odkazuje se na poznámku (note),
- `g` ... odkazuje se na glosu,
- `a` ... odkazuje se na článek (article),
- `i` ... odkazuje se na úvod (introduction).

Specifikátor odkazu se nevytiskne, je to jen interní informace, kam má směřovat aktivní odkaz (tj. proklik myši). Nenásleduje-li za ukončujícím znakem údaje `>` žádný z uvedených specifikátorů, jedná se o odkaz na verš (to je asi nejčastější případ). Příklad odkazu na poznámku:

```
... viz <"poznámku k" Jr 7:4>n, kde je uvedeno více informací.
```

vytiskne ... viz [poznámku Jr 7:4](#), kde je uvedeno více informací. Proklik vede na první poznámku k verši Jr 7:4, nikoli k verši samotnému.

V případě odkazu na článek (specifikátor `a`) má úplný údaj formát `<kniha> <kapitola>`, tj. chybí informace o verši, protože články jsou úvody ke kapitolám. V případě odkazu na úvod (specifikátor `i`) má úplný údaj formát `<kniha>` (chybí informace o kapitole i verši), protože to jsou úvody ke knihám. V ostatních případech má úplný údaj formát, jak bylo řečeno v sekci 5.1 s výjimkou popsanou v sekci 5.3

5.3 Výjimka pro formát úplného údaje u některých knih

Knihy Abd Fm 2Jn 3Jn Ju nejsou členěny na kapitoly. V odkazu na verš, poznámku k verši nebo glosu do některé z těchto knih tedy chybí informace o kapitole a formát úplného odkazu vypadá takto: `<kniha> <verš>`. Aby \TeX věděl, že má uplatnit tuto výjimku, je třeba seznam zkratek těchto knih definovat v makru `\nochapbooks`. Například v souboru `Cze-books.tex` je řečeno

```
\def\nochapbooks{Abd Fm 2Jn 3Jn Ju}
```

Protože pro různé jazyky existují různé zkraky knih, je třeba toto makro definovat v závislosti na použitém jazyku.

5.4 Neúplný údaj

Někdy dokáže umístění verše čtenář stanovit z kontextu, takže *údaj* nemusí být úplný. V *neúplném údaji* může chybět *knihy* nebo *knihy* *kapitola*: nebo všechno. Například:

```
... analogii vidíme též ve <"verších" Jr 8:13>, <9:7> a <11:3>
... viz verše <Jr 8:13>, <15>, <17>
... viz všechny <"verše" Jr 8:13--<22>,
... (srv. <Jr 8:13> a <"jeho poznámku">n).
```

Vytiskne se: ... analogii vidíme též ve [verších Jr 8:13, 9:7 a 11:3](#) ... viz verše [Jr 8:13, 15, 17](#) ... viz všechny [verše Jr 8:13–22](#). ... (srv. [Jr 8:13 a jeho poznámku](#))

Údaje 9:7, 11:3, 15, 17, 22 a poslední prázdný údaj v těchto příkladech jsou neúplné. Čtenář ví, že se týkají knihy Jeremjáš a že v případě, kdy není uvedena ani kapitola, se týkají kapitoly 8 knihy Jeremjáš. V posledním příkladě s prázdným údajem čtenář ví, že se jedná o poznámku k Jr 10:4. Ví to i \TeX a přidělí ke všem těmto neúplným odkazům správné interní odkazy (podle kterých se provede proklik), protože neúplný údaj přebírá nespecifikované informace z předchozího údaje. Toto pravidlo se vztahuje lokálně na jeden objekt textu: poznámku, článek, úvod atd. Je-li hned první údaj v některém objektu textu neúplný, nespecifikované informace se nahradí zkratkou aktuálně zpracovávané knihy, případně číslem aktuální kapitoly, případně verše.

Je-li neúplný údaj uvozen znakem \, přebírá nespecifikované informace z aktuálně zpracovávané knihy, případně kapitoly a případně verše bez ohledu, jaký údaj předchází. Například:

```
\Note 4.5 {} Myšlenka se opakuje v <Jr 4:5>. Ale také v \<8:3> najdete o tom zmínku.
```

Zde odkaz [8.3](#) vede na verš 3 kapitoly 8 aktuální knihy. Kdyby tam znak \ nebyl, pak by tento odkaz vedl na 8:3 knihy Jeremjáš.

Neúplný údaj se vytiskne tak jak je jako neúplný. Shora uvedená pravidla na jeho doplnění se provedou jen interně, aby aktivní odkaz správně fungoval po kliknutí myší.

Existuje ještě jeden odlišný formát neúplného údaje: chybí číslo verše. To se pozná tak, že je přítomná zkratka knihy a v údaji chybí dvojtečka a nejedná se o knihu ze seznamu výjimek \nochapbooks. Tedy údaj má tvar *knihy* *kapitola* (oddělené mezerou). \TeX v takovém případě vytvoří interní odkaz na první verš dané kapitoly. Příklady:

```
... viz <"kapitolu" Da 7>
... Josefův příběh (<Gn 39–41>)
```

Ve druhém příkladě je uveden rozsah kapitol a \TeX vytvoří interní odkaz na první verš kapitoly 39. Srovnajte též se sekci [5.5](#).

5.5 Formát pro rozsah veršů a pro úsek ve verši

V každém údaji je možné místo *verš* nebo místo *knihy* *kapitola*:*verš* psát rozsah veršů ve formátu *od*–*do*. \TeX vytvoří interní link jen na první verš rozsahu a spojovník (znak –, ASCII 45) uvedený v rozsahu promění v pomlčku. Příklady:

```
<Jr 8:3–7>,
<Jr 8:3–9:5>,
<3–7>,
<8:3–7>.
```

Třeba první odkaz z tohoto příkladu se vytiskne jako [Jr 8:3–7](#) a nabízí proklik jen na Jr 8:3.

Někdy je potřeba odkazovat na úsek verše, nikoli na celý verš. To se dělá připojením písmene bezprostředně za číslo verše. Například

```
... viz <Da 9:11b>
```

Takové písmeno můžete připojit k úplnému i neúplnému odkazu. Pro potřebu hyperlinku se tato písmena ignorují, ale vytisknou se. Tedy uvedený příklad vytiskne ... viz [Da 9:11b](#) ale proklik vede na Da 9:11.

5.6 Zatajení údaje

Když nepostačuje žádné z pravidel pro vytvoření interního linku z odkazu uvedené výše, můžete vše, co se má v odkazu vytisknout, uzavřít do " . . . " a následný údaj, podle kterého se má vytvořit interní odkaz, zatajit. K tomu stačí těsně za zavírací uvozovku " připojit znak podtržítko _ následovaný údajem. Tento údaj se netiskne. Například

```
<"První kniha Samuelova"_1S 1:1>
```

vytiskne pouze text [První kniha Samuelova](#), který interně odkazuje na první verš této knihy.

5.7 Přechíslování odkazu

Pokud odkaz směřuje na verš, který má v konkrétním jazykovém překladu odlišné číslování specifikované makrem \renum, pak do odkazu запиšte údaj podle výchozího číslování a \TeX si jej sám přepočítá podle dat uvedených v \renum. Vytiskne přepočítaný údaj a použije jej pro interní odkaz.

5.8 Redukce údajů

TODO Pravidlo pro tisk odkazů zní: vytiskne se přesně to, co je mezi závorkami < a > napsáno (s výjimkou případných znaků "). To je ale možné změnit. Možná budete chtít do závorek pro odkaz psát úplné údaje a požadovat jejich automatickou redukci na neúplné, odkazují-li na aktuální knihu. Neodkazují-li na aktuální knihu, údaj zůstane úplný. Toho můžete docílit přidáním \re před otevírací závorkou údaje (pravidlo případné redukce se uplatní jen na tento jediný údaj) nebo pomocí příkazu \reduceref. Když jej použijete v poznámce (nebo v \TeX ové skupině), pravidlo redukce se uplatní pro všechny následující úplné údaje této poznámky (této \TeX ové skupiny). Když jej použijete v deklaraci sazby, pravidlo případné redukce se uplatní na všechny úplné údaje v dokumentu. Příklad:

```
\re<"verš" Da 7:3>
```

se vytiskne jako [verš Da 7:3](#), je-li tento odkaz uveden mimo knihu Daniel. Když je ale tento odkaz uveden v knize Daniel, vytiskne se jen [verš 7:3](#), který interně odkazuje na Da 7:3.

Pravidlo redukce odkazů nastavené pomocí \reduceref lze vypnout příkazem \noreduceref. Od něj až do konce poznámky (\TeX ové skupiny) se odkazy chovají, jakoby \reduceref nebylo zapnuté.

5.9 Nepodařené odkazy, tj. odkazy na neexistující místo

Je-li uveden odkaz na neexistující verš nebo neexistující poznámku, pak nastanou dvě možnosti. Je-li to odkaz na knihu, která se záměrně netiskne (protože například pracujeme s testovacím výtiskem jen vybraných knih Bible, viz též \printedbooks v sekci 2.1), pak odkaz je sice obarven, jakoby byl aktivní, ale aktivní není a v logu při zpracování \TeX em se neobjeví žádné varování. Pokud ale odkaz směřuje na neexistující verš nebo poznámku tištěné části Bible, pak odkaz je aktivní, proklik směřuje na poslední stránku PDF souboru a v logu se objeví o tom varování, že odkaz není správný. Při prvním zpracování \TeX em jsou ale neošetřeny všechny odkazy a v logu je velké množství varování o nesprávných odkazech. Teprve při následujícím zpracování \TeX em jsou odkazy směřující do existujícího místa správně prolinkovány.

5.10 Trasování odkazů

Implicitně je zapnuto vypisování podrobného trasování odkazů v log souboru. Vypnete to pomocí \notracinglinks a zpětně zapnete pomocí \tracinglinks. Navíc pomocí \tracingouterlinks vypnete potlačení varování o odkazech na neexistující knihy, což umožní v logu najít případné neexistující odkazy z důvodu překlepu ve značce knihy.

5.11 Odkazy na kapitoly a na knihy

TODO Chcete-li odkázat na knihu s proklikem na začátek této knihy, pište \cref[<knihy>], například \cref[Gn] vytiskne [Gn](#) s proklikem na začátek knihy Genesis. Podobně \cref[Gn 2] vytiskne [Gn 2](#) s proklikem na začátek kapitoly Den odpočinutí. Chcete-li vytisknout něco jiného, pak za údajem [. . .] musí těsně následovat <{text}>, kde <{text}> je text, který se má vytisknout a stát se aktivním odkazem. Takže třeba \cref[Gn 2]{Den odpočinutí} vytiskne [Den odpočinutí](#) s proklikem na odpovídající kapitolu.

5.12 Odkazy na stránky

TODO Ve svém textu můžete umístit netištěný cíl odkazu na stránku pomocí `\pglabel[⟨lejblík⟩]` a pak můžete odkazovat na stránku pomocí `\pgref[⟨lejblík⟩]`. V místě `\pgref[⟨lejblík⟩]` se vytiskne číslo strany, které je možné prokliknout. Podobně jako v případě `\cref` se dá pomocí příkazu `\pgref[⟨lejblík⟩]{⟨text⟩}` vytisknout jiný `⟨text⟩` než číslo strany, který proklikne na místo, kde je umístěný `\pglabel[⟨lejblík⟩]`.

6 Varianty formátování stránek

V implicitní podobě je nastavena jednosloupcová sazba pro hlavní text Bible a pro úvody ke knihám a pro anotace. Dvousloupcová sazba je nastavena pro poznámky k veršům.

Implicitně jsou čísla kapitol velká a ve vnějším okraji. Ve vnějším okraji jsou též zvětšené značky uvozovek připojené k citacím. Příkazem `\normalchapnumbers` dojde ke změně tohoto nastavení: čísla kapitol pak jsou vsazena do levého horního rohu prvního odstavce a zvětšené značky uvozovek jsou zrušeny. Vnější okraj je pak zmenšen, protože v něm už není žádný tiskový materiál.

Další varianty formátování stránek jsou zatím jen v plánu a nejsou implementovány.

7 Možnosti hledání chyb

Může se stát, že uděláte nějaký překlep v souborech `notes-*`, `intro-*`, `articles-*`. Kdybyste soubor rovnou zařadili do celého zpracování Bible, TeX ohlásí chybu ve zcela jiném okamžiku, než když čelí příslušný soubor a zpětné dohledávání takové chyby je velmi obtížné, navíc se typicky chyba projeví v jiné fázi zpracování a tedy je oznámena často velmi nesrozumitelným a zavádějícím způsobem.

Je ovšem možné nově napsané soubory nejprve zpracovat přímo, ale bez propojení s hlavním textem a v takovém módu jsou chyby hlášeny přímočařeji. Přímé hledání chyb provedete příkazem `\checksyntax ⟨seznam souborů⟩ {}` Zde seznam souborů jsou názvy kontrolovaných souborů bez přípony `.tex`, takže například můžete mít na konci hlavního souboru:

```
\checksyntax intro-Dan articles-Dan notes-Dan {}
```

provede přímou kontrolu vyjmenovaných souborů. Na výstupu je text těchto souborů bez důrazu na jakékoli formátování. Syntaktické chyby v souborech se ale projeví přímočaře.

Poznámka: použití `\checksyntax` deaktivuje příkaz `\processbooks`, který v takovém případě neprovede nic, takže základní text Bible se vůbec nenačte.

8 Rejstřík

<code>\amark</code>	4–5	<code>\fmtadd</code>	7	<code>\processbooks</code>	4, 16
<code>\Article</code>	9–10	<code>\fmtfile</code>	4	<code>\putArticle</code>	9
<code>\articlefile</code>	4, 9	<code>\fmtins</code>	7	<code>\putCite</code>	9
<code>\begcenter</code>	7	<code>\fmtpre</code>	7	<code>\putImage</code>	8
<code>\bmark</code>	4–5	<code>\input {Cze-books.tex}</code>	3	<code>\quotedby {<autor>}</code>	10
<code>\BookException</code>	5	<code>\input {Cze-vars.tex}</code>	3	<code>\ref</code>	9
<code>\BookPost</code>	5	<code>\insertCite</code>	10	<code>\renum</code>	12
<code>\BookPre</code>	5	<code>\introfile</code>	4	<code>\swapCites</code>	10
<code>\BookTitle</code>	5	<code>\load[op-bible]</code>	3	<code>\switch</code>	12
<code>\bye</code>	4	<code>\nl</code>	10	<code>\tmark</code>	3–4, 12
<code>\chaptit{<text>}</code>	7	<code>\nochapbooks</code>	6, 13	<code>\txsfile</code>	4
<code>\checksyntax</code>	16	<code>\Note</code>	7–8, 11	<code>\variants</code>	4–5, 11
<code>\Cite</code>	10	<code>\notesfile</code>	4	<code>\vdef</code>	4–5, 10–11
<code>\cslang</code>	3	<code>\pg</code>	9	<code>\ww</code>	11
<code>\endcenter</code>	7	<code>\picdir</code>	9	<code>\x</code>	10–11
<code>\endgraf</code>	7	<code>\printedbooks</code>	4–5		