Zpravodaj Československého sdružení uživatelů TeXu

Zdeněk Wagner

Babylón mluví hindsky

Zpravodaj Československého sdružení uživatelů TeXu, Vol. 17 (2007), No. 1, 12–20

Persistent URL: http://dml.cz/dmlcz/150023

Babylón mluví hindsky

Zdeněk Wagner

Babel poskytuje jednotné rozhrani pro tvorbu vícejazyčných dokumentů. V současné době není bohužel podporován žádný z indických jazyků. Sazba v indických jazycích je založena na specializovaných balíčcích. Nejpokročilejším z nich je Velthuis Devanāgarī for TEX, protože již obsahuje hindské nadpisy i makro pro tisk data v evropském stylu. Definiční soubor pro začlenění hindštiny do babelu byl tedy v sooučasné době vyvinut.

Druhá část článku vysvětluje rozdíl mezi UNICODE a Velthuisovou transliterací. To je důležité pro pochopení nástroje sloužícího ke konverzi hindských a sanskrtských dokumentů z MS Wordu a OpenOffice.org do TEXu pomocí procesoru XSLT 2.0 a perlovského skriptu, jakož i metody, jak lze vytvořit prohledávatelný PDF soubor.

Závěrem článek diskutuje možnosti dalšího vývoje, zejména možnosti, jež nabízí XETEX a integrace jazyka LUA do pdfTEXu.

Práce byla prezentována na konferenci TUG 2006 v Marrákéši a zveřejněna v angličtině v TUGboatu [13]. Český překlad je publikován se souhlasem organizátorů konference i redakce TUGboatu.

Úvod

Balíčky pro sazbu v různých indických jazycích jak pro plain TEX, tak pro LATEX, jsou dostupné na CTAN již dlouhou dobu. Autoři těchto balíčků věnovali pozornost podpoře písma. To obsahuje potíže, jež nelze řešit TEXem samotným. Vytváření spřežek a vkládání nesamostatných samohlásek (máter) do subskriptů a superskriptů lze řešit pomocí ligatur v souborech TFM. Forma spřežky však závisí též na jazyce. Zatímco v sanskrtu a tradiční hindštině se užívají spřežky क्त (kta) a न्न (nna), v moderní hindštině jsou obvykle nahrazeny polovičními formami क्त a न्न. To lze řešit pouze preprocesorem svázaným s balíčkem (La)TEXových maker.

Balíčky používají indologové z celého světa i lidé v Indii. Je tedy nešťastné, že balíčky podporují pouze písmo a chybí jim podpora jazyka. Výjimkou je Velthuis devanāgarī for TEX [1]. Počínaje verzí 2.13 obsahuje definice nadpisů i evropského stylu data a nabízí makra umožňující přepínání mezi nimi a anglickou verzí. Nabízí tedy mini-babel, jaký obsahuje CSLATEX. Dalším přirozeným krokem je tedy vytvoření definičního souboru jazyka, jímž bude hindština integrována do Babylónu.

Stvoření definičního souboru jazyka

Cílem práce bylo umožnit transparentní použití hindštiny ve vícejazyčných dokumentech pomocí `\usepackage[hindi]{babel}`. Příprava definičního souboru jazyka pro babel není obtížná. Je nutné definovat makra s názvy nadpisů jako `\chaptername` a makro `\today` pro tisk data. Tyto definice již byly v balíčku přítomny a bylo též vyřešeno přepínání písmových variant Bombay, Calcutta a Nepali. Jejich pouhé vložení do definičního souboru jazyka však není postačující. Zobrazení textů v dévanágarském písmu vyžaduje řadu speciálních maker. Jak již bylo zmíněno dříve, hindský text nemůže být předložen TEXu přímo, ale je nutné předchozí zpracování preprocesorem. Bylo by neúčelné, kdyby makra ze souboru `devanagari.sty` byla překopírována do definičního souboru jazyka a pak musela být udržována na dvou místech. Definiční soubor jazyka tedy načítá `devanagari.sty`, jehož parametry jsou definovány jako jazykové atributy. Ve skutečnosti tento definiční soubor bez fontů a preprocesoru stejně nebude fungovat. Požadavek, že musí být instalován kompletní balíček Velthuis devanāgarī for TEX, tak nepředstavuje žádné omezení.

Balíček `devanagari.sty` obsahuje makra pro nadpisy a tisk data v angličtině. Tyto definice nelze odstranit, protože jsou dokumentovány a jejich odstranění může poškodit existující dokumenty. Na druhou stranu však tyto definice budou kolidovat s jádrem babelu. Balíček byl tedy upraven tak, aby tato makra byla definována pomocí `\providecommand` a definice je odložena použitím `\AtBeginDocument`. Tím je zajištěno, že zmíněná makra existují, ale definice z babelu mají přednost nezávisle na pořadí, v němž se balíčky načítají. Definiční soubor jazyka kontroluje verzi balíčku a stěžuje si, je-li instalována stará verze. Jak již bylo zmíněno, nestačí aktivovat hindštinu jako hlavní jazyk nebo použitím prostředí pro změnu jazyka. Text musí být uzavřen ve skupině {`\dn` ... }, jinak jej preprocesor nenajde.

UNICODE vs. Velthuisova transliterace

Devanāgarī má původ ve starém písmu Brāhmī. Patří mezi abugidy. Každá souhláska (vyanjana) obsahuje inherentní samohlásku (v písmu devanāgarī *a*) a ostatní samohlásky jsou přidávány jako diakritická znaménka v blízkosti souhlásky. Skupiny souhlásek často tvoří spřežky, jen menšina z nich se zapisuje pomocí znaku viráma. Počáteční forma samohlásek má jiný tvar než samohlásková diakritická znaménka (nesamostatné samohlásky, mātrā).

UNICODE je založen na znacích. Inherentní samohláska *a* se nezapisuje. Znak U+0915 označuje tedy slabiku क (ka). Slabika कि (ki) je reprezentována dvěma znaky UNICODE U+915 U+93F. Záměna pořadí znaků v tištěném výstupu je přenechána jako úloha zobrazovače [2]. Samostatné samohlásky

(počáteční formy) mají odlišné kódy, tj. kód इ (i) je U+0907. Trojice znaků U+0915 U+094D U+0924 označuje sanskrtskou spřežku क्त (kta). Tento znak však nemusí být přítomen v moderních fontech pro hindštinu. V takovém případě bude zobrazeno क्त. Chceme-li zobrazovač přinutit k vytvoření této formy i v případě, kdy font sanskrtskou spřežku obsahuje, musíme vložit znak zero-width-joiner. Kód v UNICODE pak bude U+0915 U+904D U+200D U+0924.

Transliterační schéma, které vyvinul Frans Velthuis, se snaží co nejvíce přiblížit školním praktikám. Devanāgarī je tradičně přepisováno latinkou, přičemž dlouhé samohlásky, cerebrální souhlásky a nosovky jsou označovány diakritickými znaménky [10]. Forma se mírně liší v různých učebnicích a slovnících. Velthuisova transliterace je sedmibitovým kódováním, takže lze znaky psát na běžné americké klávesnici. Je založena na výslovnosti, přestože v hindštině se uprostřed mnoha slov a na konci slov inherentní *a* nevyslovuje. Inherentí *a* uprostřed slov se vždy musí zapsat, ale koncové *a* obvykle vynecháváme. Slovo करना tedy musíme psát jako *karanaa*, zatímco slovo घर píšeme jen *ghar*.

Důležitost rozdílu mezi těmito přístupy bude ukázána v následujícím textu.

Konverze do Velthuisovy transliterace

Příprava knih je často společnou prací autorů, editorů a sazeče. Autoři zřídka dodají text v TEXu, často používají jiné textové editory, většinou MS Word. První úlohou je tedy konverze dodaného rukopisu do TEXu. Téměř veškeré značkování musí být odstraněno a nahrazeno jiným, které odpovídá grafickému návrhu. S výhodou lze otevřít soubor v OpenOffice.org a uložit jej v nativním formátu, jímž je XML. Ačkoli jsou dostupné různé konverzní nástroje, jejich výstup stále zachovává příliš mnoho formátování. Využití TEXML by si též vyžádalo příliš mnoho práce, kterou by stěží bylo možno použít v dalších knihách. Protože dokument v OpenOffice.org je uložen v XML, lze konverzi provést pomocí XSLT. Jednoduchý styl může odstranit veškeré formátování a zachovat tučné písmo, kurzívu a poznámky pod čarou. Některé znaky UNICODE však nejsou přímo dostupné. Naštěstí Saxon v. 8.x [5], jenž implementuje XSLT 2.0, nabízí znakové mapy, jejichž pomocí lze znaky konvertovat na TEXové sekvence. To však není pro texty v devanāgarī postačující. Pokud bychom pouze převedli znaky z UNICODE na odpovídající latinková písmena, všechna inherentní *a* by byla ztracena.

Výstup transformace XSLT lze předložit TEXu. Přesto je doporučeno jisté následné zpracování. Někdy je tučný text vložen tak, že je každé písmeno ztučněno samostatně. Spojení do jediného makra `\textbf` dosáhneme snáze v perlu než v transformačním stylu. Navíc je celý dokument v OpenOffice.org uložen na jednom řádku. Protože potřebujeme výsledný dokument editovat ručně, je vhodné je nalámat na řádky vhodné délky. Bez jakéhokoliv programování toho docílíme modulem `Text::Wrap`. Konverzi lze tedy docela snadno rozdělit mezi XSLT a perl.

Velthuisova transliterace kóduje dlouhé samohlásky buď zdvojením, nebo pomocí verzálek. Transformační styl používá vždy verzálky, aby se předešlo nejednoznačnostem. Například कई bude převedeno na *kaI*, protože *kaii* by se zobrazilo jako कैइ, což je špatně. Problém by se dal též řešit vložením prázdných složených závorek, ale použití verzálek je jednodušší. Samostatné samohlásky jsou transformovány přímo na odpovídající písmena. Před nesamostatné samohlásky je vloženo rovnítko. Souhlásky jsou následovány rovnítkem a znak viráma je transformován na podtržítko. Pak přichází na řadu perlovský skript. Zpočátku je každý odstavec samostatným řádkem, jenž musí být zpracován. První úlohou je vytvoření spřežek. To bude správně fungovat jak v sanskrtských slovech, tak v moderní hindštině, kde se některé spřežky neužívají, např. ve slově अड्डा. Znak viráma pak vloží preprocesor. Spřežky budou vytvořeny příkazem

```
while (s/(\{\\dn [^}]*)=_/$1/) {}
```

V následujícím kroku jsou doplněny nesamostatné samohlásky. Dvojice rovnítek jsou odstraněny a osamocená rovnítka označují inherentní *a*, jež musí být vložena. Je toho docíleno těmito řádky

```
while (s/(\{\\dn [^}]*)==([aAiIuU.eo])/$1$2/){}
while (s/(\{\\dn [^}]*)=/$1a/) {}
```

Pokud nekonvertujeme sanskrtský text, odstraníme ještě koncové inherentní *a*.

```
while (!$opt_sanskrit && s/(\{\\dn [^}]*)a([^.a-zA-Z])/$1$2/) {}
```

Nakonec přelámeme řádek.

```
eval {
  print wrap('', '', $_);  1;
} or do {
  warn "Warning: $@";  print;
};
```

Prohledávatelné PDF soubory

PDF soubory mají důležitou roli jako elektronické dokumenty. Názvy kapitol mohou být vloženy do záložek a příbuzné části lze provázat hypertextovými odkazy. Přesto je žádoucí, abychom byli schopni vyhledávat slova a věty. Indická písma zde přinášejí velký problém. PDF, podobně jako PostScript, zobrazuje glyfy, ale vstupní pole ve vyhledávacím dialogu očekává znaky v UNICODE. Soubory vytvořené balíčkem `devnag` jsou tedy neprohledávatelné. XeTeX [14] může používat fonty OpenType, avšak ukazuje se, že ani takto vytvořené soubory prohledávatelné nejsou. Situace není jednoduchá. XeTeX může používat různé stroje pro vytvoření PDF a výsledek je podle toho různý. OpenOffice.org je nepatrně úspěšnější, protože všechna jednoduchá slova, která neobsahují spřežky, jsou vyhledatelná.

Klíčový problém plyne z rozdílu mezi glyfy a znaky. Mapování mezi glyfy a znaky lze vložit do mapy ToUnicode. Tato vlastnost je již implementována

v balíčku `cmap.sty`. Experimentální mapa ToUnicode byla proto vytvořena pro balíček Velthuis Devanāgarī. Protože pdfTEX vkládá tyto mapy podle kódování fontu, bude každý indický fomt vyžadovat vlastní jméno kódu pro LATEX. V současnosti Devanāgarī, Bengali a Gurmukhi používají kódování U a spoléhají na preprocesor s obdobnou funkcionalitou. V tomto experimentálním projektu je definováno kódování X0900, což se vztahuje k příslušnému bloku v UNICODE.

Fonty `dvng` obsahují jednoduché znaky, spřežky a části, které se slepují dohromady TEXovými makry. Jednoduché znaky a spřežky jsou mapovány přímo na znaky UNICODE. Poloviční formy souhlásek jsou mapovány na příslušné znaky následované virámem. Vattu je přidáváno k úplné souhlásce. Je proto mapováno na र (ra), před nímž je přidán znak viráma. Tím jsou znaky se spřežkami vyhledatelné.

Všechny problémy bohužel vyřešit nelze. Protože PDF pracuje s glyfy, jsou nesamostatné samohlásky *i* vloženy před souhlásky. Vyhledáváme-li taková slova, je nutno zapsat je do vstupního pole ve vyhledávacím dialogu tak, jak vypadají graficky, tj. musíme psát nesamostatné *i* před souhláskou. Acrobat Reader je často zmaten samohláskou umístěnou nad nebo pod souhláskou i znakem *vattu*. Vytvoří se tím nadbytečná mezislovní mezera. Hledáme-li tedy कुल्लू, musíme do vstupního pole zapsat dvě slova कु ल्लू. Slovo ड्राइवर musíme zapsat způsobem, jenž je ve Velthuisově systému nemožný, a to jako dvě slova ड्र इवर. UNICODE povoluje zapsat na začátek slova nesamostatnou samohlásku, což je, samozřejmě, nesprávné. Díky této chybné vlastnosti jsou uvedená slova vyhledávatelná. Nadbytečná mezera se nevytvoří, pokud je slabika kreslena jako jeden glyf ve fontu. Slova रुकना a मात्रा lze najít v podobě, jak jsou zapsána. Podrobnější informace jsou uvedeny v dokumentaci experimentální mapy ToUnicode [12].

Mapa ToUnicode se používá též při kopírování textu ze souboru PDF do jiné aplikace. Je tedy pochopitelné, že nastávají stejné komplikace. Vložený text má nadbytečné mezery za znakem vattu a za krátkou i dlouhou nesamostatnou samohláskou *u*. Slova s krátkým nesamostatným *i* budou vizuálně vypadat dobře, ale jejich reprezentace v UNICODE bude špatná. Pokud slovo दिल्ली zkopírované z PDF bude posláno do řadicího algoritmu, objeví se na nesmyslném místě, protože program najde slovo začínající nesamostatnou samohláskou *i*, což hindský pravopis nepřipouští.

Budoucí vývoj

Největší nevýhodou současného balíčku Velthuis devanāgarī for TEX je nutnost použití preprocesoru. Pokud připravujete dokument v jediném jazyce, nemusí to připadat tak nepohodlné. Sázíte-li však trojjazyčný dokument v hindštině, bengálštině a pandžábštině, musíte zpracovat zdrojový soubor postupně třemi preprocesory. Preprocesor DEVNAG umí zpracovat několik pandžábských a bengál-

ských slov v hindském odstavci použitím úhlových závorek, ale všechny preprocesory nemají tak pokročilé schopnosti. Z nutnosti použití preprocesoru vyplývají těžkosti při tvorbě rejstříku, jež nebyly dosud vyřešeny. Náhrada preprocesoru jiným mechanismem je tedy důležitým krokem.

První myšlenkou byla reimplementace preprocesoru v encTEXu [6]. Ten umí konvertovat vstupní znaky na libovolné tokeny podle konverzní tabulky. Konverzi lze zapínat a vypínat změnou hodnoty `\mubytein`. Konverzní tabulka může též ovlivnit zápis do souboru primitivem `\write` podle hodnoty v registru `\mubyteout`. Protože konverze probíhá na znakové úrovni ve vstupním procesoru, nelze odlišit znaky ve slovech od znaků v řídicích sekvencích. Reimplementace preprocesoru v encTEXu by tedy nebyla snadná a výsledný kód by nebyl efektivní.

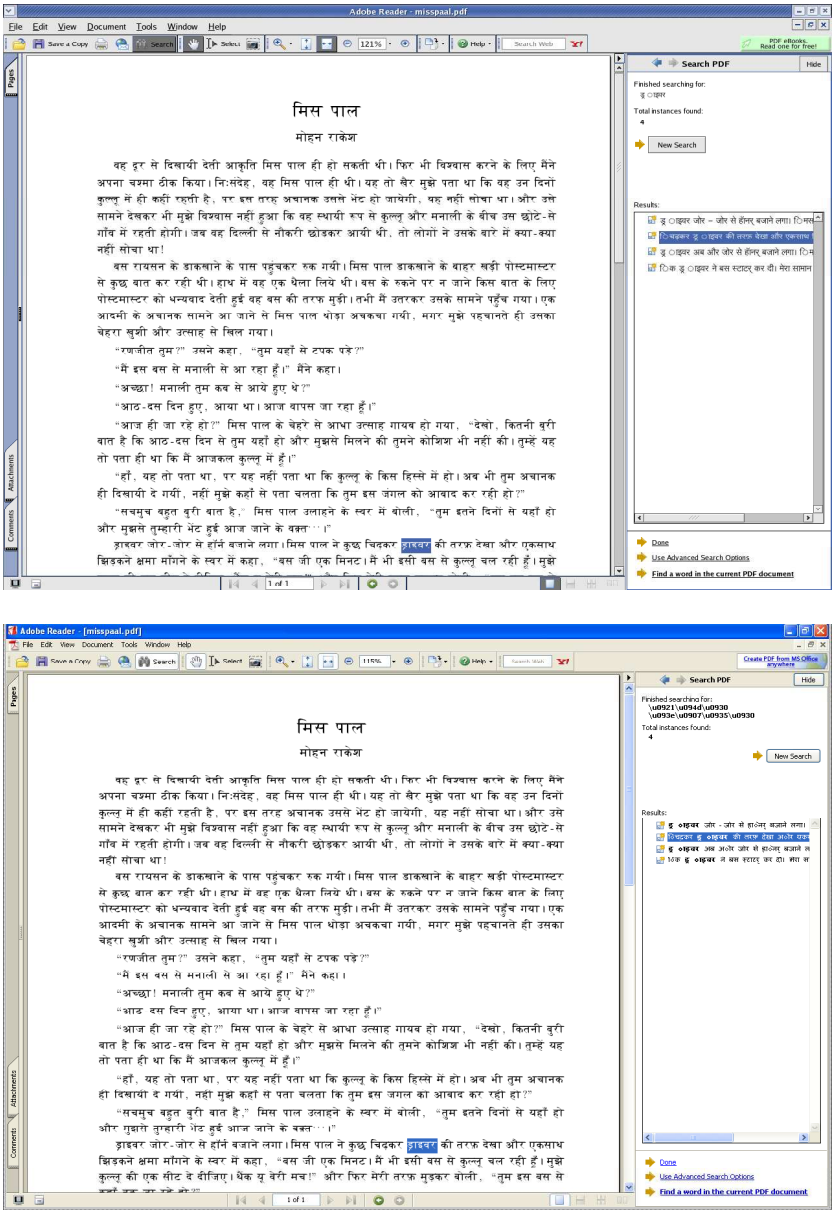
Slibnější možnost nabízí integrace skriptovacího jazyka LUA [3] do pdfTEXu. Preprocesor lze reimplementovat v jazyku LUA, a navíc lze přidat další vylepšení. Bude možno číst texty jak ve Velthuisově transliteraci, tak v UNICODE. Bude též možno vysázet dokumenty, které již byly zpracovány preprocesorem, takže kompatibilita nebude ztracena. Jakmile bude LUA integrován do XETEXu, bude možno volit mezi fonty `dvng` a OpenType. Pak bude též snazší implementovat software pro tvorbu rejstříků v indických jazycích.

Požadavky na vícejazyčné prostředí

Vícejazyčné prostředí není vyžadováno jen v TEXu, ale v celém operačním systému. Především je nutné zobrazit správně všechny znaky UNICODE. Skupiny souhlásek s virámy je nutno správně spojit do spřežek a nesamostatné samohlásky přesunout na správná místa. V Linuxu k tomu slouží například knihovny ICU [4] a Pango [8]. Tyto knihovny však nejsou dosud používány všemi programy. Ani Firefox nepoužívá Pango ve standardní konfiguraci, ale musí být použití aktivováno nastavením `MOZ_PANGO_ENABLE=1`. MS Internet Explorer zobrazuje hindské texty korektně.

Zdrojový text musí též být připraven v UNICODE. Zde je stále problém s editory v Linuxu. OpenOffice.org, gEdit a <oXygen/> (XML editor) [7] pracují správně. Nezkoušel jsem yudit, ale podle referencí jiných uživatelů je prý též s indickými jazyky funkční. Podpora indických písem v jiných editorech stále chybí.

Vstupní text v indických písmech musí být přijímán a správně zobrazován ve všech aplikacích. Zde má problém Adobe Acrobat Reader ve Windows. Pomocí hindské klávesnice se mi nepodařilo napsat nic do vstupního pole vyhledávacího dialogu. Sice se cosi podivného zobrazilo, ale hledání vůbec nefungovalo. Text může být přenesen z jiné aplikace pomocí copy&paste, ale zobrazí se v entitách UNICODE. Srovnání verzí pro Linux a Windows je uvedeno na obrázku 1. Verze pro Linux je uživatelsky pohodlnější, ale obě vyhledávají stejně dobře.



Obrázek 1: Vyhledávání slova ड्राइवर (vloženého jako ड् ◻ इवर) v Adobe Acrobat Readeru 7 v Linuxu (nahoře) a ve Windows XP (dole).

K plné funkčnosti vyhledávání a kopírováni textů v PDF je nutno upravit specifikaci CMap. V současnosti jsou možná mapování 1:1 a 1:mnoho. Abychom mohli měnit pořadí glyfů a správně seřadit vícedílné samohlásky v drávidských jazycích, potřebujeme mapování mnoho:mnoho. Pro lepší pochopení je v tabulce 1 ukázáno několik dévanágarských a malajálamských slabik.

Tabulka 1: Vybrané dévanágarské a malajálamské slabiky

Význam	**Devanāgarī** देवनागरी	**Malayālam** മലയാളം
ma	म	മ
maa	मा	മാ
mi	मि	മി
mii	मी	മീ
me	मे	മെ
mo	मो	മോ

Závěr

V článku je ukázáno, jak byl vytvořen babelovský modul pro hindštinu. Dále se autor zamýšlí nad možnostmi použití indických textů v elektronických dokumentech. Jsou též popsány nástroje pro konverzi souborů z MS Wordu a OpenOffice.org do TEXu a pro tvorbu prohledávatelných souborů PDF. Nástroje jsou dostupné z autorovy webové stránky [11].

Poděkování

Autor děkuje ostatním vývojářům Velthuis Devanāgarī for TEX, protože zmíněný balíček je základem tohoto projektu, jakož i Karlu Píškovi za konverzi indických fontů do formátu Type 1 [9]. Zvláštní poděkování zaslouží Anshuman Pandey za překlad nadpisů do hindštiny a John Smith a Arnošt Štědrý za poskytnutí testovacích souborů vytvořených XƎTEXem. Autor dále děkuje Alexandru Babičovi za provedení testů v Ubuntu a Petru Tomáškovi za vysvětlení způsobu zobrazení písem v Xorg. Za finanční podporu účasti na konferenci TUG 2006 autor děkuje CSTUGu a TUGu.

Reference

[1] Devanāgarī for TEX. `http://devnag.sarovar.org/`.

[2] Joan Aliprand et al. *The Unicode Standard*, chapter South Asian Scripts. The Unicode Consortium, 2003. http://www.unicode.org/faq/indic.html#5.
[3] Hans Hagen. LuaTEX: Howling to the moon. *TUGboat*, 26(2):152–157, 2005. http://www.tug.org/TUGboat/Contents/contents26-2.html.
[4] International components for Unicode. http://icu.sourceforge.net/.
[5] Michael Kay. Saxon, the XSLT and XQuery processor. http://saxon.sourceforge.net/.
[6] Petr Olšák. encTEX. http://www.olsak.net/enctex.html.
[7] <oXygen/> xml editor. http://www.oxygenxml.com/.
[8] Pango. http://www.pango.org/.
[9] Karel Píška. Indic Type 1 fonts for TEX. CTAN:fonts/ps-type1/indic.
[10] Transliteration pages. http://homepage.ntlworld.com/stone-catend/translit.htm.
[11] Zdeněk Wagner. Můj volně šiřitelný software. http://icebearsoft.euweb.cz/sw.php.
[12] Zdeněk Wagner. Prohledávatelné pdf s dévanágarskými texty. http://icebearsoft.euweb.cz/dvngpdf/.
[13] Zdeněk Wagner. Babel speaks Hindi. *TUGboat*, 27(2):176–180, 2006.
[14] The XƎTEX typesetting system. http://scripts.sil.org/xetex.

Summary: Babel Speaks Hindi

Babel provides unified interface for creation of multilingual documents. Unfortunately none of Indic languages is currently supported. Typesetting in Indic languages is based on specialised packages. The most advanced from them is Velthuis Devanāgarī for TEX because it already provides Hindi captions as well as a macro for a Europian style date. A language definition file for plugging Hindi into babel has therefore been recently developed.

The second part of the paper explains differences between UNICODE and Velthuis transliteration. This is important for understanding the tool that can convert Hindi and Sanskrit documents from MS Word and OpenOffice.org into TEX via an XSLT 2.0 processor and a Perl script as well as a method of making the PDF files searchable.

Finally the paper discusses some possibilities of further development, mainly the advantages offered by XƎTEX and by forthcoming integration of LUA into pdfTEX.