Mongolian Cyrillic Support for XeLaTeX and LuaLaTeX

Apostolos Syropoulos
Xanthi, Greece
asyropoulos@yahoo.com

Bat-erdene Altangerel
Ulaanbaatar, Mongolia
echingiss@gmail.com

2019/12/14

Abstract

The xecyrmongolian package provides basic support for Mongolian Cyrillic so to be able to prepare documents with either XeLaTeXor LuaLaTeX.

1 Introduction

The package xecyrmongolian has been designed for people who want to prepare documents whose main language is Mongolian Cyrillic and want to typeset their work with either XeLaTeXor LuaLaTeX. The package allows users to load other hyphenation patterns so to be able to create trully multilingial documents. In addition, all standard enumerations use the Cyrillic alphabet used in Mongolia. The following simple LaTeX code is a typical usage example of the package.

```
\documentclass[a4paper]{article}
\usepackage{xltxtra}
\usepackage{enumitem}
\usepackage{xecyrmongolian}
\begin{document}
  \setmainfont{MenkGarqagTig.ttf} % or any font you like
  \begin{enumerate}[label=(\Alph*)]
     \item an apple
     \item a banana
     \item a carrot
     \item a durian
  \end{enumerate}
  \Useg{12}
\end{document}
```

2 The Source Code

First we define the various strings that correspond to the standard LaTeX captions.

```
⟨*xecyrmongolian⟩
\message{Package `xecyrmongolian' version 1.0.0 by ^^J%
         Apostolos Syropoulos and Bat-erdene Altangerel}
\def\prefacename{Оршил}%
\def\refname{Ашигласан ном}%
\def\abstractname{Товчлол}%
\def\bibname{Ашигласан номзүй}%
\def\chaptername{Бүлэг}%
\def\appendixname{Хавсралт}%
```

```
\def\contentsname{Гарчиг}%
\def\listfigurename{Зургийн жагсаалт}%
\def\listtablename{Хүснэгтийн жагсаалт}%
\def\indexname{Товъёог}%
\def\figurename{Зураг}%
\def\tablename{Хүснэгт}%
\def\partname{Хэсэг}%
\def\enclname{Оруулах}%
\def\ccname{Мэдэгдэл}%
\def\headtoname{}%
\def\pagename{Хуудас}%
\def\seename{Үзнэ үү}%
\def\alsoname{мөн үзнэ үү}%
\def\proofname{Баталгаа:}%
\def\glossaryname{Тайлбар}%
```

\Useg
\useg

Next, we define the macros \Useg and \useg that are the Mongolian counterpart of \Alph and \alph, respectively. However, these commands should not be used in enumerations, etc. It is better to make the \Alph commands to produce Cyrillic letters by giving the command \usegalph. The behavior of this command is the default behavior of the package.

```
\def\Useg#1{\ifcase#1\or
    А\or Б\or В\or Г\or Д\or Е\or Ё\or Ж\or З\or
    И\or Й\or К\or Л\or М\or Н\or О\or Ө\or П\or
    Р\or С\or Т\or У\or Ү\or Ф\or Х\or Ц\or Ч\or
    Ш\or Щ\or Ъ\or Ы\or Ь\or Э\or Ю\or Я\else\@ctrerr\fi}
\def\useg#1{\ifcase#1\or
    а\or б\or в\or г\or д\or е\or ё\or ж\or з\or
    и\or й\or к\or л\or м\or н\or о\or ө\or п\or
    р\or с\or т\or у\or ү\or ф\or х\or ц\or ч\or
    ш\or щ\or ъ\or ы\or ь\or э\or ю\or я\else\@ctrerr\fi}
```

The previous commands do not work if their argument is a counter. And since we may want to use them in enumeration or to number chapters, we introduce the following commands that work properly when their arguments are counters.

```
\def\useg@mong#1{\expandafter\useg\expandafter{\the#1}}
\def\Useg@mong#1{\expandafter\Useg\expandafter{\the#1}}
```

\mongmonth

Now we redefine \today so as to produce dates in Mongolian. The names of months are defined by the macro \mongmonth.

```
\def\mongmonth{%
  \ifcase\month\or 1-р ~сарын\or 2-р ~сарын\or 3-р ~сарын\or 4-р ~сарын\or
  5-р ~сарын\or 6-р ~сарын\or 7-р ~сарын\or 8-р ~сарын\or 9-р ~сарын\or
  10-р ~сарын\or 11-р ~сарын\or 12-р ~сарын\fi}
\def\today{\number\year~оны~\mongmonth\space \number\day}
```

LuaLaTeX and XeLaTeX have different ways to load hyphenation patterns. Package luahyphenrules by Javier Bezos facilitates this process for people who want to use LuaLaTeX and the "traditional" way to load hyphenation patterns. To ensure proper inclusion of LuaTeX staff, I use the following "idiom":

\ifx\directlua\undefined *non LuaLaTeX code*\else *LuaLaTeX code*\fi

```
\ifx\directlua\undefined\else\RequirePackage{luahyphenrules}\fi
```

The code that follows loads the hyphenation patterns. The XeLaTeX code is quite standard and depends on the babel pattern loading mechanism, while the LuaLaTeX code uses the \HyphenRules macro, which has essentially the functionality of the \selectlanguage macro.

```
\ifx\directlua\undefined%
   \language\l@mongolian\else\HyphenRules{mongolian}\fi%
```

By default the Mongolian alphabetic enumeration is used instead of enumerations with Latin letters.

```
\let\latin@alph\@alph
\let\latin@Alph\@Alph
\let\@alph\useg@mong
\let\@Alph\Useg@mong
```

`\nousegalph` `\usegalph` If for some reason, the user needs to have the original enumeration back, then the user should used the command `\nousegalph`. And if she wants to switch back, then she has to use the `\usegalph` command:

```
\def\nousegalph{%
  \let\@alph\latin@alph
  \let\@Alph\latin@Alph}
\def\usegalph{%
  \let\@alph\useg@mong
  \let\@Alph\Useg@mong}
```

`\setlanguage` We provide the `\setlanguage` command which activates the hypehnation patterns of some other language. It is similar to babel's `\selectlanguage`, but we opted to use a new name to avoid possible name conflicts. Valid arguments include `monogreek`, `mongolian`, and `american`. As was noted previously, package luahyphenrules provides the command `\HyphenRules` which has exactly the same functionality as this command. So when using LuaLaTeX users will actually use the `\HyphenRules` command. And since the main language of the documenyt will be Mongolian, we have to load the Mongolian hypehnation patterns.

```
\ifx\directlua\undefined%
  \def\setlanguage#1{%
     \expandafter\ifx\csname l@#1\endcsname\relax%
     \typeout{^^J Error: No hyphenation pattern for language #1 loaded,}%
     \typeout{ default hyphenation patterns are used.^^J}%
     \language=0%
     \else\language=\csname l@#1\endcsname\fi}
\else
  \let\setlanguage\HyphenRules
\fi
\setlanguage{mongolian}
```

⟨/xecyrmongolian⟩