The Bulgarian language definition file for the babel system*

Version v1.2g

Georgi N. Boshnakov†

2013/10/14

Съдържание

1 Introduction **2**

2 User guide **2**
2.1 Settings things up . . . 2
2.1.1 Settings for LaTeX and pdfLaTeX . . . 3
2.1.2 Settings for Lualatex . . . 4
2.2 Switching to and from Bulgarian language . . . 4
2.2.1 The most common case . . . 4
2.2.2 Total change of language . . . 5
2.2.3 Compatibility commands for change of language . . . 5
2.3 Quotation marks . . . 6
2.4 Hyphens and dashes . . . 6
2.5 Hyphenation control . . . 7
2.6 Other shorthands . . . 7
2.7 Dates . . . 8
2.8 Enumerations . . . 8
2.9 Mathematical functions . . . 9
2.10 Troubleshooting . . . 9

3 History **9**
3.1 Changes in version 1.2 . . . 9
3.2 Changes in version 1.1 . . . 9
3.3 Changes in version 1.0a–1.0g . . . 9
3.4 Version 1.0 . . . 10

*The file described in this section has version number v1.2g and was last revised on 2013/10/14.

†The original, non-babel, version (bulgaria.sty) was developed by Georgi N. Boshnakov in 1994–1996. It was later adapted for babel by Georgi N. Boshnakov and Johannes Braams using as a template the August-1998 version of `russianb.dtx` and heavily borrowing code from there. I am also grateful to Vladimir Volovich for his helpful and timely advice. For versions 1.1-1.2 of the Bulgarian style, significant additions and modification to the code were made and the user manual was almost completely rewritten.

3.5 Older versions 10

4 Implementation **10**
4.1 Font encodings 10
4.1.1 Adjustments for X2 encoding 12
4.2 Input encoding 13
4.3 Switching between Cyrillic and Latin encodings 14
4.4 Captions for Bulgarian 14
4.4.1 Captions for pre3.9 babel 14
4.4.2 Captions for babel 3.9 or later 15
4.5 Dates in Bulgarian 17
4.5.1 Dates for pre3.9 babel 17
4.5.2 Dates for babel 3.9 or later 18
4.6 Extras for Bulgarian 19
4.7 Enumerations for Bulgarian 22
4.8 Cyrillic letters in maths 24
4.9 Alternative names for math functions 24
4.10 Compatibility with older versions 25
4.11 Finish 25

1 Introduction

This file provides Bulgarian language support for the babel system.

This version of the Bulgarian style is the first public revision afer the release of version 3.9 of babel but it should be compatible with previous versions of babel. More specifically, documents that compile successfully with the last version of babel before 3.9 should continue to compile. The documentation is draft, suggestions and meaningful examples are welcome.

This version corrects some bugs and introduces support for Luatex and XeTeX. The documentation has been rewritten (to a large extent). A few other small changes have been made as well, see the index of changes at the end of this document for details.

Please report any bugs to the current maintainer (Georgi N. Boshnakov).

2 User guide

2.1 Settings things up

The Bulgarian style described in this document is used by supplying `bulgarian` as one of the language options to babel when loading it. If more than one language is requested, the last one designates the main language of the document and it will be active after `begin{document}`. For example, the command

```
\usepackage[english,bulgarian]{babel}
```

loads babel with the English and Bulgarian styles and sets Bulgarian as the main language.

For recent versions of babel the main language can be specified also with the help of the option `main`:

```
\usepackage[main=bulgarian,english]{babel}
```

For typesetting with the classic engines, LaTeX and pdfLaTeX, it is also necessary to specify the encoding of the source file (input encoding) and the font encoding. For the unicode based engines, LuaLaTeX and Xetex, this is not necessary but but one needs to specify fonts. These issues are briefly considered below.

2.1.1 Settings for LaTeX and pdfLaTeX

Here is a template to be inserted towards the beginning of the main LaTeX file, after the `documentclass` command:

```
\usepackage[T1,T2A]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage[bulgarian]{babel}
```

This assumes UTF-8 encoded file. If this is not the case, `utf8` should be replaced by the actual encoding, most likely `cp1251`:

```
\usepackage[T1,T2A]{fontenc}
\usepackage[cp1251]{inputenc}
\usepackage[bulgarian]{babel}
```

If the input encoding is not specified (i.e. the corresponding line above is missing), a warning is issued in the log file and LaTeX still tries to typeset the document. It may be sucessful but it is better to specify the encoding explicitly.

The options for `fontenc` may look misterious but it is safe to ignore the details and simply insert the above command as is. Basically, `T1` and `T2A` can be considered standard TeX encodings for the Latin and Cyrillic letters, respectively. `T2A` contains Latin letters, as well, so `\usepackage[T2A]{fontenc}` is often good enough.

Actually, the encodings have some fallback defaults and a document may compile successfully even if one or both of `fontenc` and `inputenc` are omitted. However, it is a good idea to supply explicit `inputenc` in any case.

The support provided by `inputenc` package for unicode symbols is not complete. If LaTeX complains about undefined or unknown unicode symbols, there is the alternative to use package `ucs` and specify `utf8x` for inputenc, as in:

```
\usepackage{ucs}
\usepackage[utf8x]{inputenc}
```

In many cases this simply works. Otherwise see the documentation of package `ucs` for further details.

2.1.2 Settings for Lualatex

Lualatex requires that the source file is in UTF-8 encoding. It requires also the specification of fonts. This is done by loading package `fontspec` (attention: not `fontenc`) and issuing a sequence of font setting commands. The following can be used to arrange for the unicode versions of the traditional Computer Roman fonts to be used in the document:

```
\usepackage{fontspec}
\setmainfont{CMU Serif}
\setsansfont{CMU Sans Serif}
\setmonofont{CMU Typewriter Text}
\usepackage[english,bulgarian]{babel}
```

Any system fonts can be used instead. For example, the main font command above could be replaced by the following:

```
\setmainfont[Ligatures=TeX]{Times New Roman}
```

Package `fontspec` offers a lot of flexibility for the font selection, see its documentation.

Compatibility mode Lualatex can be set up to use 8-bit fonts if the Unicode fonts do not work for some reason. Note that the source file still needs to be in UTF-8! Here is a template for this (the difference from LaTeX is in the second line):

```
\usepackage[T1,T2A]{fontenc}
\usepackage[lutf8]{luainputenc}
\usepackage[english,bulgarian]{babel}
```

The compatibility mode is a remedy for emergencies, not for routine use.

2.2 Switching to and from Bulgarian language

The language at the beginning of the document, i.e. after `\begin{document}`, is determined by the options used to load package babel. Several ways of changing the language are available, depending on what the user needs.

2.2.1 The most common case

In the most common case the user just wants to typeset some piece of text in a different language, using the typesetting and hyphenation rules of that language, without changing generic strings like `\today`, names of sections and things like `\proofname` to the chosen language. This can be done with the command `\foreignlanguage` or the environment `otherlanguage*`.

Here are some unimaginative examples to illustrate this. Here is a phrase in Bulgarian „тази дата е форматирана по американския начин: October 14, 2013“, typeset in a sentence written in English. This phrase was typeset as follows:

```
\foreignlanguage{bulgarian}{''‘тази дата е форматирана
 по американския начин: \today’''}
```

For longer pieces of text it is more convenient to use the equivalent environment.

```
\begin{otherlanguage*}{bulgarian}
   Това е пример за американския начин за форматиране на дати: \today.
   А тук думата \proofname е отпечатана с командата |\proofname|.
\end{otherlanguage*}
```

Notice the '*' in the name of the environment. The result is:

> Това е пример за американския начин за форматиране на дати: October 14, 2013. А тук думата Proof е отпечатана с командата `\proofname`.

Notice that the date and the heading for a proof are still in English. To change these and similar language dependent strings we need a total change of language. This is discussed below.

2.2.2 Total change of language

The babel system controls a number of typesetting features for each language. The command `\selectlanguage` is used to do a complete change to a new language. For example, `\selectlanguage{bulgarian}` switches every aspect of the babel system to Bulgarian language. For example, the following snippet:

```
\selectlanguage{bulgarian}
  Този документ е компилиран на \today.

  Думата ’’‘\proofname’’’{} е отпечатана с командата |\proofname|.
\selectlanguage{english}
Some English text...
```

produces:

> Този документ е компилиран на 14 октомври 2013 г..
>
> Думата „Доказателство“ е отпечатана с командата `\proofname`.
> Some English text...

Compare the result here to the similar example in the previous section. Now everything is changed, including `\today` and other names.

2.2.3 Compatibility commands for change of language

Version 1.1 of the Bulgarian style continues to provide a number of additional commands for change of language. They are kept mainly for compatibility with older versions of the Bulgarian style and their use is discouraged.

The command `\Bulgarian` changes only the hyphenation patterns and the encoding. It does not make a complete change of language and is really suitable for short fragments which do not contain special features. This may cause trouble in some cases, e.g. with `aux` files. Similarly, the command `\English` switches to the English language. It is left for compatibility.

For historical and compatibility reasons with ancient version of the Bulgarian style, there are also several abbreviations equivalent to `\Bulgarian`. These are: `\Bul`, `\Bg`, `\cyrillictext`, `\cyr`. Also, `\selectbglanguage` is equivalent to `\selectlanguage{bulgarian}`.

Similarly, `\Eng` is equivalent to `\English` and `\selectenglanguage` is equivalent to `\selectlanguage{english}`.

2.3 Quotation marks

Quotation marks with typographic quality used to be a problem in the past. They were not available on typewrites and keyboards. Moreover, the opening quotes used in Bulgarian typesetting were not available in the standard fonts and needed to be generated by other means. With modern computers, fonts and editing programs it is no longer a problem to enter directly the correct opening and closing quotation marks, especially when Unicode based encodigs are in use.

Nevertheless, commands and shorthands are convenient. The special commands for quotation marks available for the Bulgarian language are shown in Table 1. The German quotes are the current norm. The French ones can be found mostly

Command	Shorthand	Meaning
`\glqq`	`"``	for German left double quotes (looks like„).
`\grqq`	`"'`	for German right double quotes (looks like “).
`\flqq`	`"<`	for French left double quotes (looks like ≪).
`\frqq`	`">`	for French right double quotes (looks like ≫).
`\dq`		the original quotes character (`"`).

Таблица 1: Special quotes available for Bulgarian typesetting

in older texts.

The French quotes are also available as ligatures ‘`<<`’ and ‘`>>`’ in 8-bit Cyrillic font encodings (`LCY`, `X2`, `T2*`) and as ‘`<`’ and ‘`>`’ characters in 7-bit Cyrillic font encodings (`OT2` and `LWN`).

2.4 Hyphens and dashes

The em-dash in Bulgarian texts is used for several purposes. Table 2 shows the available commands and their intended contexts. The second column shows also the shorthands that are usually used to enter these dashes.

Command	Shorthand	Meaning
`\cdash---`	`"---`	Cyrillic emdash in plain text.
`\cdash--~`	`"--~`	Cyrillic emdash in compound names (surnames).
`\cdash--*`	`"--*`	Cyrillic emdash for denoting direct speech.

Таблица 2: The emdashes available for Bulgarian typesetting

LaTeX source	Typeset output
`П. П. Славейков`	П. П. Славейков
`П.'',П.'',Славейков`	П. П. Славейков

Таблица 3: Example for initials spaced out by ordinary space (first line) and the special command (second line). The command leaves a smaller, more visually appealing space. It also allows hyphenation of the surname, although this is not visible here.

2.5 Hyphenation control

Table 4 gives commands for hyphenation control. Most of them aim to facilitate hyphenation in compound words and related circumstances.

Shorthand	Meaning
`''-`	an explicit hyphen sign, allowing hyphenation in the rest of the word.
`''"`	like `''-`, but producing no hyphen sign (for compound words with hyphen, e.g. `x-''"y` or some other signs as "disable/enable").
`''~`	for a compound word mark without a breakpoint.
`''=`	for a compound word mark with a breakpoint, allowing hyphenation in the composing words.
`'',`	thinspace for initials with a breakpoint in following surname.

Таблица 4: Extra definitions, mainly for hyphenation control

Hyphenating words containing a hyphen is not particularly desirable but may be necessary not only for long compounds words like министър-председател (prime minister), but also for comparative and superlative adjectives like по-красноречива (more eloquent) or най-безопасният (the safest), since the Bulgarian equivalents of 'more' and 'most' are connected to the adjective by a hyphen. Even so, when the line width is not short, which is the case for this document, TeX is usually able to avoid hyphenating such words. In this paragraph one of the example words is hyphenated only because I deliberately edited it until this happened.

2.6 Other shorthands

Table 5 gives some additional shorthands. Currently there is only one entry and it concerns ligatures. Ligatures are not common in Bulgarian texts. A ligature is used to replace a sequence of two or more symbols by a more visually appealing glyph. This topic is not of much importance for typesetting Bulgarian texts and is beyond the scope of this document. For example, two dashes, '`--`', normally form a ligature to produce an en-dash, '–'. The command `''|` provides a way to prevent the formation of the ligature. So, '`-''|-`' gives '--'.

Shorthand	Meaning
’’|	disable ligature at this position.

Таблица 5: Extra shorthands.

2.7 Dates

The command `\today` prints the current date. It is in Bulgarian if that is the active language, see also Section 2.2. Alternatively, the command `\todayRoman` prints the current date using Roman numerals for months. So, when this document is compiled, `\today` and `\todayRoman` give „14 октомври 2013 г.“ and „14. X. 2013 г.“, respectively.

Note: Following a suggestion by Boyko Bantchev, the space between the year and ‘г.’ was reduced in version 1.2b of this style.

The command `\weekdaynamebulgarian` can be used to typeset the names of the days of the week. It is mostly intended for use in macros and has one argument, the number of the required day of the week. For example, `\weekdaynamebulgarian{1}` and `\weekdaynamebulgarian{7}` give понеделник and неделя, respectively.

2.8 Enumerations

The traditional alphabetical enumerations in Bulgarian texts use the Cyrillic alphabet (bar several letters). In principle, enumerations are a matter for class and style designers but the same can be said also about things, other than enumerations, such as names of sections and bibliography lists.

The Bulgarian style by default turns on enumeration with Cyrillic letters. This means that enumerated lists that would be labelled with Latin letters in Latin scripts are labelled with Cyrillic ones instead.

This automatic feature may not always be desirable, so Version 1.1d of the Bulgarian style streamlines somewhat the support for Cyrillic enumerations. For compatibility with older documents, the default is still to turn on the Bulgarian enumerations when the active language is Bulgarian but facilities to control this are provided.

The Cyrillic enumeration can also be turned on and off selectively for parts of the document using the commands `\abvon` and `\abvoff`. The command `\abvoff` turns off the Cyrillic enumeration. The command `\abvon` turns it on. It is thus possible to turn the feature on and off for parts of the document.

The user can turn off the Cyrillic enumeration by using attribute ‘abvoff’ when loading babel, e.g.

```
\usepackage[english,bulgarian .abvoff]{babel}
```

The dot in front of the attribute’s name is required and tells babel that this is an attribute.

2.9 Mathematical functions

The Bulgarian style provides definitions for some mathematical functions whose names differ from the corresponding English names.

Currently these functions are: `\tg`, `\ctg`, `\cosec`, `\arctg`, `\arcctg`, `\sh`, `\ch`, `\cth`.

2.10 Troubleshooting

The standard font encoding handling for Cyrillic letters in non-Uniceode encodings is based on definitions like `\CYRA` for the Cyrillic letters. These are not necessary for unicode engines (Luatex, XeTeX) but may still be present in other packages (for example `varioref`) that handle different languages by storing hard coded strings.

If your document is using a Unicode encoding, most likely UTF-8, and you get errors like „undefined command `\CYRA`“, then the command `\cyrxtounicode` may be used as an emergency patch. It is not to be used routinely as its use may hide unrelated bugs. Just put it at the beginning of your document.

3 History

3.1 Changes in version 1.2

The user visible changes are in the documentation.

The minor version number was changed from 1 to 2 (hence 1.2 rather than 1.1) to accomodate a change in babel 3.9g which removed the command `UseStrings` and replaced it with a starred version of `\StartBabelCommands`.

3.2 Changes in version 1.1

This is the first release after the individual languages were taken out of the core babel system (babel version 3.9f). Some of the most visible changes:

- Support for LuaTeX and XeTeX;
- Documentation rewritten;
- Facilities for turning on and off automatic Bulgarian enumeration;
- Some long standing bugs fixed. In particular, it should no longer be necessary to load `amsmath` before babel.

For more details on the changes, see the „Change history“ index at the end of this document.

3.3 Changes in version 1.0a–1.0g

Bug fixes and small changes by Johannes Braams.

3.4 Version 1.0

Johannes Braams modified (and improved) the original draft to conform to the babel system and incorporated `bulgarian.dtx` into it.

3.5 Older versions

The first draft of the Bulgarian style for babel was created by modifying the August-1998 version of `russianb.dtx` for the Bulgarian language along the lines of the 1994/1996 (non-babel) Bulgarian style (`bulgaria.sty`) by Georgi N. Boshnakov. It is (reasonably) backward compatible with that style—files prepared for that style should compile successfully (with vastly improved appearance due to usage of standard fonts).

„

4 Implementation

The macro `\LdfInit` takes care of preventing that this file is loaded more than once, checking the category code of the `@` sign, etc.

```
1 ⟨*code⟩
2 \LdfInit{bulgarian}{captionsbulgarian}
```

When this file is read as an option, i.e., by the `\usepackage` command, `bulgarian` will be an 'unknown' language, in which case we have to make it known. So we check for the existence of `\l@bulgarian` to see whether we have to do something here.

```
3 \ifx\l@bulgarian\@undefined
4   \@nopatterns{Bulgarian}
5   \adddialect\l@bulgarian0
6 \fi
```

New in version 1.1: detect luatex or xetex. The code is taken from russinab.dtx but the counter is renamed to avoid possible conflicts.

```
7 \newif\if@bul@unicode
8 \ifdefined\luatexversion \@bul@unicodetrue \else
9 \ifdefined\XeTeXrevision \@bul@unicodetrue \fi\fi
```

4.1 Font encodings

`\latinencoding` We need to know the fond encoding for text that is supposed to be active at the end of the babel package. This is effectively assumed to be a Latin encoding and the macro `\latinencoding`, defined by babel, contains the name of the Latin encoding.

`\cyrillicencoding` We need also to determine the encoding for Cyrillic text. It is normally loaded by the `fontenc` package. These days it is usually `T2A`.

We parse the `\cdp@list` containing the encodings known to LaTeX in the order they were loaded. We set the `\cyrillicencoding` to the *last* loaded encoding in the list of supported Cyrillic encodings: `OT2`, `LWN`, `LCY`, `X2`[1], `T2C`, `T2B`, `T2A`, if any.

```
\def\reserved@a#1#2{%
  \edef\reserved@b{#1}%
  \edef\reserved@c{#2}%
  \ifx\reserved@b\reserved@c
    \let\cyrillicencoding\reserved@c
  \fi}
\def\cdp@elt#1#2#3#4{%
  \reserved@a{#1}{OT2}%
  \reserved@a{#1}{LWN}%
  \reserved@a{#1}{LCY}%
  \reserved@a{#1}{X2}%
  \reserved@a{#1}{T2C}%
  \reserved@a{#1}{T2B}%
  \reserved@a{#1}{T2A}}
\cdp@list
```

If `\cyrillicencoding` is undefined, then the user did not load any of the supported encodings. The code below is new in version 1.1 of this file and considers also the active TeX engine. It was inspired by russianb.dtx. We set a sensible default `\cyrillicencoding` but still issue a warning to alert the user that a default font encoding is used.

```
\ifx\cyrillicencoding\undefined
  \if@bul@unicode
    \ifdefined\XeTeXrevision
      \edef\cyrillicencoding{EU1}
    \else\ifdefined\luatexversion
      \edef\cyrillicencoding{EU2}
    \fi\fi
  \else
    \edef\cyrillicencoding{T2A}
  \fi
  \PackageWarning{bulgarian.ldf}%
    {No Cyrillic font encoding has been loaded so far.\MessageBreak
     A font encoding should be declared before babel.\MessageBreak
     Default `\cyrillicencoding' encoding will be loaded
    }%
```

We avoid `\usepackage[\cyrillicencoding]{fontenc}` because we don't want to force the switch of `\encodingdefault`.

```
  \lowercase\expandafter{\expandafter\input\cyrillicencoding enc.def\relax}%
\fi

\PackageInfo{babel}
    {Using `\cyrillicencoding' as a default Cyrillic encoding}%
```

[1]Encoding `X2` does not contain Latin letters and users should be very careful to switch the language every time they want to typeset a Latin word inside a Bulgarian phrase or vice versa.

TODO: add `\English` to the extras for Bulgarian. It should not be defined globally as it may be defined at a more appropriate place elsewhere. In any case, the value of these abbreviations is limited as they do not do a complete switch of language.

The following commands are shorthands for switching to Bulgarian and English. They are present here mainly for compatibility with older versions of the Bulgarian style.

`\Bulgarian` A simple switch to Bulgarian language:

```
\DeclareRobustCommand{\Bulgarian}{%
  \fontencoding\cyrillicencoding\selectfont
  \let\encodingdefault\cyrillicencoding
  \expandafter\set@hyphenmins\bulgarianhyphenmins
  \language\l@bulgarian}
```

`\English` A simple switch, similar to the above, but to the English language:

```
\DeclareRobustCommand{\English}{%
  \fontencoding\latinencoding\selectfont
  \let\encodingdefault\latinencoding
  \expandafter\set@hyphenmins\englishhyphenmins
  \language\l@english}
```

`\Bul` `\Bg` `\cyrillictext` `\cyr` `\Eng` `\selectenglanguage` `\selectbglanguage` Finally, a few shorthands for switching the languages. These are mostly remnants from pre-babel times.

```
\let\Bul\Bulgarian
\let\Bg\Bulgarian
\let\cyrillictext\Bulgarian
\let\cyr\Bulgarian
\let\Eng\English
\def\selectenglanguage{\selectlanguage{english}}
\def\selectbglanguage{\selectlanguage{bulgarian}}
```

4.1.1 Adjustments for X2 encoding

The code in this section is executed only if the font encoding is `X2`

Since the `X2` encoding does not contain Latin letters, we should make some redefinitions of LaTeX macros which implicitly produce Latin letters.

```
\expandafter\ifx\csname T@X2\endcsname\relax\else
```

We put `\latinencoding` in braces to avoid problems with `\@alph` inside minipages (e.g., footnotes inside minipages) where `\@alph` is expanded and we get for example '`\fontencoding OT1`' (`\fontencoding` is robust).

```
\def\@Alph@eng#1{{\fontencoding{\latinencoding}\selectfont \ifcase#1\or
    A\or B\or C\or D\or E\or F\or G\or H\or I\or J\or K\or L\or M\or
    N\or O\or P\or Q\or R\or S\or T\or U\or V\or W\or X\or Y\or Z\else
    \@ctrerr\fi}}%
\def\@alph@eng#1{{\fontencoding{\latinencoding}\selectfont \ifcase#1\or
    a\or b\or c\or d\or e\or f\or g\or h\or i\or j\or k\or l\or m\or
```

```
    n\or o\or p\or q\or r\or s\or t\or u\or v\or w\or x\or y\or z\else
    \@ctrerr\fi}}%
  \let\@Alph\@Alph@eng
  \let\@alph\@alph@eng
```

Unfortunately, the commands `\AA` and `\aa` are not encoding dependent in LaTeX (unlike e.g., `\oe` or `\DH`). They are defined as `\r{A}` and `\r{a}`. This leads to unpredictable results when the font encoding does not contain the Latin letters 'A' and 'a' (like `X2`).

```
  \DeclareTextSymbolDefault{\AA}{OT1}
  \DeclareTextSymbolDefault{\aa}{OT1}
  \DeclareTextCommand{\AA}{OT1}{\r A}
  \DeclareTextCommand{\aa}{OT1}{\r a}
\fi
```

4.2 Input encoding

For LaTeX, pdfLaTeX, and compatibility mode of LuaLaTeX, the user should use the `inputenc` package (before loading babel) to specify the encoding of the input file[2]. For the Unicode engines this is not necessary, as they require Unicode based encoding.

We issue a warning if the input encoding has not been specified but do not consider this an error.

```
\@ifpackageloaded{inputenc}{%
  \if@bul@unicode
   \PackageWarning{bulgarian.ldf}{inputenc should not be used with LuaTeX or XeTeX}
  \fi
}{%
  %\def\reserved@a{LWN}%
  %\ifx\reserved@a\cyrillicencoding\else
    \def\reserved@a{OT2}%
    \ifx\reserved@a\cyrillicencoding\else
       \def\reserved@a{EU1}%
       \ifx\reserved@a\cyrillicencoding\else
          \def\reserved@a{EU2}%
          \ifx\reserved@a\cyrillicencoding\else
             \PackageWarning{bulgarian.ldf}%
                {No input encoding specified for Bulgarian language}
    \fi\fi\fi
  %\fi
}
```

[2]After the changes in version 1.1b of `bulgarian.dtx`, if the font and input encoding are `T2A` and `cp1251`, respectively, the document might be processed correctly even without the `inputenc` package. Even so, it is better to be explicit about the input encoding.

\cyrillictext
\latintext
\lat

4.3 Switching between Cyrillic and Latin encodings

The command \cyrillictext will switch from Latin font encoding to the Cyrillic font encoding, the command \latintext switches back. We add the shorthand \lat for \latintext. This assumes that the 'normal' font encoding is a Latin one. These commands are *declarations*, for shorter peaces of text the commands \textlatin and \textcyrillic can be used.

\latintext is defined in the core of babel, while \cyrillictext is defined above.

```
\let\lat\latintext
```

\textcyrillic
\textlatin

These commands take an argument which is then typeset using the requested font encoding. \textlatin is defined in the core of babel. (It is defined there with \DeclareRobustCommand instead.)

```
\DeclareTextFontCommand{\textcyrillic}{\cyrillictext}
```

4.4 Captions for Bulgarian

If \bbl@version is undefined we use the old approach (using \addto commands).

```
\ifx\bbl@version\@undefined
\PackageInfo{bulgarian.ldf}{Executing the pre 3.9 branch for captions}
```

4.4.1 Captions for pre3.9 babel

\captionsbulgarian

The macro \captionsbulgarian defines all strings used in the four standard document classes provided with LaTeX. The two commands \cyr and \lat activate Cyrillic, resp. Latin, encoding.

```
\addto\captionsbulgarian{%
  \def\prefacename{%
    {\cyr\CYRP\cyrr\cyre\cyrd\cyrg\cyro\cyrv\cyro\cyrr}}%
  \def\refname{%
    {\cyr\CYRL\cyri\cyrt\cyre\cyrr\cyra\cyrt\cyru\cyrr\cyra}}%
  \def\abstractname{%
    {\cyr\CYRA\cyrb\cyrs\cyrt\cyrr\cyra\cyrk\cyrt}}%
  \def\bibname{%
    {\cyr\CYRB\cyri\cyrb\cyrl\cyri\cyro\cyrg\cyrr\cyra\cyrf\cyri\cyrya}}%
  \def\chaptername{%
    {\cyr\CYRG\cyrl\cyra\cyrv\cyra}}%
  \def\appendixname{%
    {\cyr\CYRP\cyrr\cyri\cyrl\cyro\cyrzh\cyre\cyrn\cyri\cyre}}%
  \def\contentsname{%
    {\cyr\CYRS\cyrhrdsn\cyrd\cyrhrdsn\cyrr\cyrzh\cyra\cyrn\cyri\cyre}}%
  \def\listfigurename{%
    {\cyr\CYRS\cyrp\cyri\cyrs\cyrhrdsn\cyrk\ \cyrn\cyra\
      \cyrf\cyri\cyrg\cyru\cyrr\cyri\cyrt\cyre}}%
  \def\listtablename{%
    {\cyr\CYRS\cyrp\cyri\cyrs\cyrhrdsn\cyrk\ \cyrn\cyra\
```

```
      \cyrt\cyra\cyrb\cyrl\cyri\cyrc\cyri\cyrt\cyre}}%
  \def\indexname{%
    {\cyr\CYRA\cyrz\cyrb\cyru\cyrch\cyre\cyrn\
      \cyru\cyrk\cyra\cyrz\cyra\cyrt\cyre\cyrl}}%
  \def\authorname{%
    {\cyr\CYRI\cyrm\cyre\cyrn\cyre\cyrn\
      \cyru\cyrk\cyra\cyrz\cyra\cyrt\cyre\cyrl}}%
  \def\figurename{%
    {\cyr\CYRF\cyri\cyrg\cyru\cyrr\cyra}}%
  \def\tablename{%
    {\cyr\CYRT\cyra\cyrb\cyrl\cyri\cyrc\cyra}}%
  \def\partname{%
    {\cyr\CYRCH\cyra\cyrs\cyrt}}%
  \def\enclname{%
    {\cyr\CYRP\cyrr\cyri\cyrl\cyro\cyrzh\cyre\cyrn\cyri\cyrya}}%
  \def\ccname{%
    {\cyr\cyrk\cyro\cyrp\cyri\cyrya}}%
  \def\headtoname{%
    {\cyr\CYRZ\cyra}}%
  \def\pagename{%
    {\cyr\CYRS\cyrt\cyrr.}}%
  \def\seename{%
    {\cyr\cyrv\cyrzh.}}%
  \def\alsoname{%
    {\cyr\cyrv\cyrzh.\ \cyrs\cyrhrdsn\cyrshch\cyro\ \cyri}}%
  \def\proofname{%
    {\cyr\CYRD\cyro\cyrk\cyra\cyrz\cyra\cyrt\cyre\cyrl\cyrs\cyrt\cyrv\cyro}}%
  \def\glossaryname{%
    {\cyr\CYRP\cyrr\cyri\cyrt\cyru\cyrr\cyrk\cyra}}% Glossary: Притурка, Гло-
cap?
}%
```

4.4.2 Captions for babel 3.9 or later

```
\else
\PackageInfo{bulgarian.ldf}{Executing the post 3.9 branch for captions}
\StartBabelCommands*{bulgarian}{captions}[unicode, fontenc=EU1 EU2, charset=utf8]
  \SetString\prefacename{Предговор}
  \SetString\refname{Литература}
  \SetString\abstractname{Абстракт}
  \SetString\bibname{Библиография}
  \SetString\chaptername{Глава}
  \SetString\appendixname{Приложение} % todo: Апендикс ?
  \SetString\contentsname{Съдържание}
  \SetString\listfigurename{Списък на фигурите}
  \SetString\listtablename{Списък на таблиците}
  \SetString\indexname{Азбучен указател}
  \SetString\authorname{Именен указател}
  \SetString\figurename{Фигура}
  \SetString\tablename{Таблица}
```

```
  \SetString\partname{Част}
  \SetString\enclname{Приложения}
  \SetString\ccname{копия}
  \SetString\headtoname{За}
  \SetString\pagename{Стр.}  % todo: shouldn't this be all lowercase?
  \SetString\seename{вж.}
  \SetString\alsoname{вж. също и}
  \SetString\proofname{Доказателство}
  \SetString\glossaryname{Притурка}

\StartBabelCommands*{bulgarian}{captions}
  \SetString\prefacename{%
    {\cyr\CYRP\cyrr\cyre\cyrd\cyrg\cyro\cyrv\cyro\cyrr}}%
  \SetString\refname{%
    {\cyr\CYRL\cyri\cyrt\cyre\cyrr\cyra\cyrt\cyru\cyrr\cyra}}%
  \SetString\abstractname{%
    {\cyr\CYRA\cyrb\cyrs\cyrt\cyrr\cyra\cyrk\cyrt}}%
  \SetString\bibname{%
    {\cyr\CYRB\cyri\cyrb\cyrl\cyri\cyro\cyrg\cyrr\cyra\cyrf\cyri\cyrya}}%
  \SetString\chaptername{%
    {\cyr\CYRG\cyrl\cyra\cyrv\cyra}}%
  \SetString\appendixname{%
    {\cyr\CYRP\cyrr\cyri\cyrl\cyro\cyrzh\cyre\cyrn\cyri\cyre}}%
  \SetString\contentsname{%
    {\cyr\CYRS\cyrhrdsn\cyrd\cyrhrdsn\cyrr\cyrzh\cyra\cyrn\cyri\cyre}}%
  \SetString\listfigurename{%
    {\cyr\CYRS\cyrp\cyri\cyrs\cyrhrdsn\cyrk\ \cyrn\cyra\
      \cyrf\cyri\cyrg\cyru\cyrr\cyri\cyrt\cyre}}%
  \SetString\listtablename{%
    {\cyr\CYRS\cyrp\cyri\cyrs\cyrhrdsn\cyrk\ \cyrn\cyra\
      \cyrt\cyra\cyrb\cyrl\cyri\cyrc\cyri\cyrt\cyre}}%
  \SetString\indexname{%
    {\cyr\CYRA\cyrz\cyrb\cyru\cyrch\cyre\cyrn\
      \cyru\cyrk\cyra\cyrz\cyra\cyrt\cyre\cyrl}}%
  \SetString\authorname{%
    {\cyr\CYRI\cyrm\cyre\cyrn\cyre\cyrn\
      \cyru\cyrk\cyra\cyrz\cyra\cyrt\cyre\cyrl}}%
  \SetString\figurename{%
    {\cyr\CYRF\cyri\cyrg\cyru\cyrr\cyra}}%
  \SetString\tablename{%
    {\cyr\CYRT\cyra\cyrb\cyrl\cyri\cyrc\cyra}}%
  \SetString\partname{%
    {\cyr\CYRCH\cyra\cyrs\cyrt}}%
  \SetString\enclname{%
    {\cyr\CYRP\cyrr\cyri\cyrl\cyro\cyrzh\cyre\cyrn\cyri\cyrya}}%
  \SetString\ccname{%
    {\cyr\cyrk\cyro\cyrp\cyri\cyrya}}%
  \SetString\headtoname{%
    {\cyr\CYRZ\cyra}}%
  \SetString\pagename{%
```

```
    {\cyr\CYRS\cyrt\cyrr.}}%
  \SetString\seename{%
    {\cyr\cyrv\cyrzh.}}%
  \SetString\alsoname{%
    {\cyr\cyrv\cyrzh.\ \cyrs\cyrhrdsn\cyrshch\cyro\ \cyri}}%
  \SetString\proofname{%
    {\cyr\CYRD\cyro\cyrk\cyra\cyrz\cyra\cyrt\cyre\cyrl\cyrs\cyrt\cyrv\cyro}}%
  \SetString\glossaryname{%
    {\cyr\CYRP\cyrr\cyri\cyrt\cyru\cyrr\cyrk\cyra}}% Glossary: Притурка, Гло-
  cap?
\EndBabelCommands
\fi
```

4.5 Dates in Bulgarian

The month is often written with Roman numbers in Bulgarian dates. Below we define a version of `\today` in this format but here we define a couple of potentially useful general commands.

`\month@Roman` `\Romannumeral` `\Romannumeral` converts its argument to a capitalized Roman numeral. `\month@Roman` gives the current month using capital Roman numerals.

```
\def\month@Roman{\expandafter\@Roman\month}%
\def\Romannumeral#1{\uppercase\expandafter{\romannumeral #1}}
%%% 2013-10-02 comment this out (it is repeated below):
%%% \def\todayRoman{\number\day.\Romannumeral{\month}.\number\year\,\cyrg.}
```

Define the dates of the switch from Julian to Gregorian calendar in Bulgaria.

`\lastJulianDatebulgarian` `\firstGregorianDatebulgarian` `\weekdaynamebulgarian`

```
\def\lastJulianDatebulgarian{19160331}
\def\firstGregorianDatebulgarian{19160414}
\def\weekdaynamebulgarian#1{\csname weekday\romannumeral#1 name\endcsname}
```

4.5.1 Dates for pre3.9 babel

We use the new mechanism provided by babel v3.9 but for backward compatibility, at least until version v3.9 of babel becomes ubiquitous we keep old code. The macro `\bbl@version` is undefined in babel versions prior to 3.9, so we use it.

If `\bbl@version` is undefined we use the old approach

```
\ifx\bbl@version\@undefined
\PackageInfo{bulgarian.ldf}{Executing the pre 3.9 branch for dates}
```

`\datebulgarian` The macro `\datebulgarian` redefines the command `\today` to produce Bulgarian dates. It also provides the command `\todayRoman` which produces the date with the month in capital roman numerals, a popular format for dates in Bulgarian.

```
\def\datebulgarian{%
  \def\month@bulgarian{\ifcase\month\or
    \cyrya\cyrn\cyru\cyra\cyrr\cyri\or
    \cyrf\cyre\cyrv\cyrr\cyru\cyra\cyrr\cyri\or
```

```
    \cyrm\cyra\cyrr\cyrt\or
    \cyra\cyrp\cyrr\cyri\cyrl\or
    \cyrm\cyra\cyrishrt\or
    \cyryu\cyrn\cyri\or
    \cyryu\cyrl\cyri\or
    \cyra\cyrv\cyrg\cyru\cyrs\cyrt\or
    \cyrs\cyre\cyrp\cyrt\cyre\cyrm\cyrv\cyrr\cyri\or
    \cyro\cyrk\cyrt\cyro\cyrm\cyrv\cyrr\cyri\or
    \cyrn\cyro\cyre\cyrm\cyrv\cyrr\cyri\or
    \cyrd\cyre\cyrk\cyre\cyrm\cyrv\cyrr\cyri\fi}%
  \def\abbgyear{\cyrg.}

  \def\today{\number\day~\month@bulgarian\ \number\year\,\abbgyear}%
  \def\todayRoman{\number\day.\,\month@Roman.\,\number\year\,\abbgyear}%
}%
```

4.5.2 Dates for babel 3.9 or later

```
\else
\PackageInfo{bulgarian.ldf}{Executing the post 3.9 branch for dates}
\StartBabelCommands*{bulgarian}{date}[unicode, fontenc=EU1 EU2, charset=utf8]
  \SetStringLoop{month#1name}{%
    януари,февруари,март,април,май,юни,%
    юли,август,септември,октомври,ноември,декември}

  \SetStringLoop{weekday#1name}{%
    понеделник,вторник,сряда,четвъртък,петък,събота,неделя}

  \SetString\abbgyear{г.}
```

The following `\if` clause is a patch. `\if@bul@unicode` ensures that the following will not be executed Unicode engines are in use. It should not be anyway but `\SetStringLoop` seems to expand its argument prematurely and `\cyrya` etc. are not defined for Unicode engines.

```
\StartBabelCommands*{bulgarian}{date}
%%% %(princ (to-cyrx-string bulmonths))
\if@bul@unicode
\else
  \SetStringLoop{month#1name}{%
    \cyrya\cyrn\cyru\cyra\cyrr\cyri,\cyrf\cyre\cyrv\cyrr\cyru\cyra\cyrr\cyri,%
    \cyrm\cyra\cyrr\cyrt,\cyra\cyrp\cyrr\cyri\cyrl,\cyrm\cyra\cyrishrt,%
    \cyryu\cyrn\cyri,\cyryu\cyrl\cyri,\cyra\cyrv\cyrg\cyru\cyrs\cyrt,%
    \cyrs\cyre\cyrp\cyrt\cyre\cyrm\cyrv\cyrr\cyri,%
    \cyro\cyrk\cyrt\cyro\cyrm\cyrv\cyrr\cyri,%
    \cyrn\cyro\cyre\cyrm\cyrv\cyrr\cyri,\cyrd\cyre\cyrk\cyre\cyrm\cyrv\cyrr\cyri}%
%%% %(princ (to-cyrx-string ”понеделник,вторник,сряда,четвъртък,петък,събота,неделя”))
  \SetStringLoop{weekday#1name}{%
    \cyrp\cyro\cyrn\cyre\cyrd\cyre\cyrl\cyrn\cyri\cyrk,%
    \cyrv\cyrt\cyro\cyrr\cyrn\cyri\cyrk,\cyrs\cyrr\cyrya\cyrd\cyra,%
    \cyrch\cyre\cyrt\cyrv\cyrhrdsn\cyrr\cyrt\cyrhrdsn\cyrk,%
```

```
    \cyrp\cyre\cyrt\cyrhrdsn\cyrk,\cyrs\cyrhrdsn\cyrb\cyro\cyrt\cyra,%
    \cyrn\cyre\cyrd\cyre\cyrl\cyrya}%
\fi

  \SetString\abbgyear{\cyrg.}

 \SetString\today{\number\day~%
   \csname month\romannumeral\month name\endcsname\space
   \number\year\,\abbgyear}

 \SetString\todayRoman{\number\day.\,\month@Roman.\,\number\year\,\abbgyear}
\EndBabelCommands
\fi
```

4.6 Extras for Bulgarian

\extrasbulgarian The macro \extrasbulgarian will perform all the extra definitions needed for the Bulgarian language. The macro \noextrasbulgarian is used to cancel the actions of \extrasbulgarian.

The first action we define is to switch on the selected Cyrillic encoding whenever we enter 'bulgarian'.

```
\addto\extrasbulgarian{\cyrillictext}
```

When the encoding definition file was processed by LaTeX the current font encoding is stored in \latinencoding. We switch back to \latinencoding whenever the Bulgarian language is no longer 'active'.

```
\addto\noextrasbulgarian{\latintext}
```

For Bulgarian, the '' character is made active. The user part of the documentation gives an overview of the available shorthands, see Table 1.

```
\initiate@active@char{''}
```

We specify that the Bulgarian group of shorthands should be used.

```
\addto\extrasbulgarian{\languageshorthands{bulgarian}}
```

These characters are 'turned on' once, later their definition may vary.

```
\addto\extrasbulgarian{%
  \bbl@activate{''}}
\addto\noextrasbulgarian{%
  \bbl@deactivate{''}}
```

To be able to define the function of '"', we first define a couple of 'support' macros.

\dq We save the original double quote character in \dq to keep it available, the math accent \''can now be typed as '"'.

```
\begingroup \catcode`\''12
\def\reserved@a{\endgroup
  \def\@SS{\mathchar''7019}
  \def\dq{''}}
\reserved@a
```

Now we can define the doublequote macros: german and french quotes. We use definitions of these quotes made in babel.sty. The french quotes are contained in the `T2*` encodings.

```
\declare@shorthand{bulgarian}{''`}{\glqq}
\declare@shorthand{bulgarian}{'''}{\grqq}
\declare@shorthand{bulgarian}{''<}{\flqq}
\declare@shorthand{bulgarian}{''>}{\frqq}
```

Some additional commands:

```
\declare@shorthand{bulgarian}{'''}{\hskip\z@skip}
\declare@shorthand{bulgarian}{''~}{\textormath{\leavevmode\hbox{-}}{-}}
\declare@shorthand{bulgarian}{''=}{\nobreak-\hskip\z@skip}
\declare@shorthand{bulgarian}{''|}{% |
\textormath{\nobreak\discretionary{-}{}{\kern.03em}%
\allowhyphens}{}}
```

The next two macros for `''-` and `''---` are somewhat different. We must check whether the second token is a hyphen character:

```
\declare@shorthand{bulgarian}{''-}{%
```

If the next token is '-', we typeset an emdash, otherwise a hyphen sign:

```
  \def\bulgarian@sh@tmp{%
    \if\bulgarian@sh@next-\expandafter\bulgarian@sh@emdash
    \else\expandafter\bulgarian@sh@hyphen\fi
  }%
```

TEX looks for the next token after the first '-': the meaning of this token is written to `\bulgarian@sh@next` and `\bulgarian@sh@tmp` is called.

```
  \futurelet\bulgarian@sh@next\bulgarian@sh@tmp}
```

Here are the definitions of hyphen and emdash. First the hyphen:

```
\def\bulgarian@sh@hyphen{\nobreak\-\bbl@allowhyphens}
```

For the emdash definition, there are the two parameters: we must 'eat' two last hyphen signs of our emdash ...:

```
\def\bulgarian@sh@emdash#1#2{\cdash-#1#2}
```

`\cdash` ... these two parameters are useful for another macro: `\cdash`:

```
\ifx\cdash\undefined % should be defined earlier
\def\cdash#1#2#3{\def\tempx@{#3}%
\def\tempa@{-}\def\tempb@{~}\def\tempc@{*}%
 \ifx\tempx@\tempa@\@Acdash\else
  \ifx\tempx@\tempb@\@Bcdash\else
   \ifx\tempx@\tempc@\@Ccdash\else
    \errmessage{Wrong usage of cdash}\fi\fi\fi}
```

second parameter (or third for `\cdash`) shows what kind of emdash to create in next step

`''---` ordinary (plain) Cyrillic emdash inside text: an unbreakable thinspace will be inserted before only in case of a *space* before the dash (it is necessary for dashes after display maths formulae: there could be lists, enumerations etc. started with "—where a is ..." i.e., the dash starts a line). (Firstly there were planned rather soft rules for user:he may put a space before the dash or not. But it is difficult to place this thinspace automatically, i.e., by checking modes because after display formulae TEX uses horizontal mode. Maybe there is a misunderstanding? Maybe there is another way?) After a dash a breakable thinspace is always placed;

```
\def\@Acdash{\ifdim\lastskip>\z@\unskip\nobreak\hskip.2em\fi
\cyrdash\hskip.2em\ignorespaces}%
```

`''--~` emdash in compound names or surnames (like Mendeleev–Klapeiron); this dash has no space characters around; after the dash some space is added by `\exhyphenpenalty`.

```
\def\@Bcdash{\leavevmode\ifdim\lastskip>\z@\unskip\fi
 \nobreak\cyrdash\penalty\exhyphenpenalty\hskip\z@skip\ignorespaces}%
```

`''--*` for denoting direct speech (a space like `\enskip` must follow the emdash);

```
\def\@Ccdash{\leavevmode
 \nobreak\cyrdash\nobreak\hskip.35em\ignorespaces}%
%\fi
```

`\cyrdash` Finally the macro for "body" of the Cyrillic emdash. The `\cyrdash` macro will be defined in case this macro hasn't been defined in a fontenc file. For T2*fonts, cyrdash will be placed in the code of the English emdash thus it uses ligature `---`.

```
% Is there an IF necessary?
\ifx\cyrdash\undefined
\def\cyrdash{\hbox to.8em{--\hss--}}
\fi
```

Here a really new macro—to place thinspace between initials. This macro used instead of `\,` allows hyphenation in the following surname.

```
\declare@shorthand{bulgarian}{'',}{\nobreak\hskip.2em\ignorespaces}
```

The Bulgarian hyphenation patterns can be used with `\lefthyphenmin`[3] and `\righthyphenmin` set to 2.

```
\providehyphenmins{\CurrentOption}{\tw@\tw@}
\fi
```

Bulgarian typesetting requires `frenchspacing`. So, we add commands to `\extrasbulgarian` and `\noextrasbulgarian` to turn it on and off, respectively.

```
\addto\extrasbulgarian{\bbl@frenchspacing}
\addto\noextrasbulgarian{\bbl@nonfrenchspacing}
```

[3] Actually, it seems that the „official" definition allows even one character for `lefthyphen` but I have not investigated this completely.

4.7 Enumerations for Bulgarian

`\@Alph@bul` We begin by defining `\@Alph@bul` which works like `\@Alph`, but produces (uppercase) Cyrillic letters intead of Latin ones. The letters й, ъ and ы (ISHRT, HRDSN and SFTSN) are skipped, as usual for this kind of enumeration. Note that these macros do not switch encodings (and never did)[4].

```
\if@bul@unicode
  \def\@Alph@bul#1{\ifcase#1\or
    А\or Б\or В\or Г\or Д\or Е\or Ж\or
    З\or И\or К\or Л\or М\or Н\or О\or
    П\or Р\or С\or Т\or У\or Ф\or Х\or
    Ц\or Ч\or Ш\or Щ\or Ю\or Я\else
    \@ctrerr\fi}
\else
  \def\@Alph@bul#1{\ifcase#1\or
    \CYRA\or \CYRB\or \CYRV\or \CYRG\or \CYRD\or \CYRE\or \CYRZH\or
    \CYRZ\or \CYRI\or \CYRK\or \CYRL\or \CYRM\or \CYRN\or \CYRO\or
    \CYRP\or \CYRR\or \CYRS\or \CYRT\or \CYRU\or \CYRF\or \CYRH\or
    \CYRC\or \CYRCH\or \CYRSH\or \CYRSHCH\or \CYRYU\or \CYRYA\else
    \@ctrerr\fi}
\fi
```

`\@alph@bul` The macro `\@alph@bul` is similar to `\@Alph@bul` but produces lowercase Bulgarian letters.

```
\if@bul@unicode
  \def\@alph@bul#1{\ifcase#1\or
    а\or б\or в\or г\or д\or е\or ж\or
    з\or и\or к\or л\or м\or н\or о\or
    п\or р\or с\or т\or у\or ф\or х\or
    ц\or ч\or ш\or щ\or э\or ю\or я\else\@ctrerr\fi}
\else
  \def\@alph@bul#1{\ifcase#1\or
    \cyra\or \cyrb\or \cyrv\or \cyrg\or \cyrd\or \cyre\or \cyrzh\or
    \cyrz\or \cyri\or \cyrk\or \cyrl\or \cyrm\or \cyrn\or \cyro\or
    \cyrp\or \cyrr\or \cyrs\or \cyrt\or \cyru\or \cyrf\or \cyrh\or
    \cyrc\or \cyrch\or \cyrsh\or \cyrshch\or \cyryu\or \cyrya\else
    \@ctrerr\fi}
\fi
```

`\@Alph@eng` `\@alph@eng` We no longer define English `\@Alph@eng` and `\@alph@eng` from scratch. We copy the definitions active at the time this file is loaded.

```
\let\@Alph@eng\@Alph
\let\@alph@eng\@alph
```

[4] Earlier in this file, there are definitions of `@Alph@eng` and `@alph@eng` in the code for handling the obsolete cyrillic encoding `X2`. They contain encoding commands. This is necessary for `X2` since it does not contain Latin letters. These precautions hardly had any effect before the changes for versions 1.1, since the old code below was effectively overwriting them. The changes to `Alph` and `alph` are now honoured as a side effect of the changes.

For version 1.1d we store the original definitions in the following macros. TODO: there is redundancy but need to check before removing stuff.

```
\let\@Alph@saved\@Alph
\let\@alph@saved\@alph
```

The old code for defining `\@Alph@eng` and `\@alph@eng` is commented out.

```
%%\def\@Alph@eng#1{%
%%  \ifcase#1\or
%%  A\or B\or C\or D\or E\or F\or G\or H\or I\or J\or K\or L\or M\or
%%  N\or O\or P\or Q\or R\or S\or T\or U\or V\or W\or X\or Y\or Z\else
%%  \@ctrerr\fi
%%  }
%%\def\@alph@eng#1{%
%%  \ifcase#1\or
%%  a\or b\or c\or d\or e\or f\or g\or h\or i\or j\or k\or l\or m\or
%%  n\or o\or p\or q\or r\or s\or t\or u\or v\or w\or x\or y\or z\else
%%  \@ctrerr\fi
%%  }
%%
```

We define commands for manually switching to and from English based enumeration. Note that this definition for `\enumBul` is really suitable for switching back after using `\enumEng`, since `\enumBul` does not specify encoding and therefore would give an error if called when the active encoding does not know cyrillic (but see the similar code for X2 which does take care of the encoding).

```
\def\enumBul{\let\@Alph\@Alph@bul \let\@alph\@alph@bul}
\def\enumEng{\let\@Alph\@Alph@eng \let\@alph\@alph@eng}
\def\enumLat{\let\@Alph\@Alph@eng \let\@alph\@alph@eng}
```

`\abvon` `\abvoff` Commands are provided for manual switching on and off of the enumeration with Cyrillic letters: `\abvon` turns it on, while `\abvoff` turns it off.

```
\def\abvon{\let\@Alph@saved\@Alph \let\@alph@saved\@alph \enumBul}
\def\abvoff{\let\@Alph\@Alph@saved \let\@alph\@alph@saved}
```

By default the Bulgarian enumeration scheme is turned on when switching to Bulgarian. todo: Could avoid adding to `noextrasbulgarian` by using `babel@save` in the first command. TODO: there should be a better way to do this but I do not know of a portable way to check if the user has supplied the attribute 'abvoff'. (Attributes are processed by babel after it is loaded.)

```
\def\autoabvon{\abvon}
\def\autoabvoff{\abvoff}
\addto\extrasbulgarian{\autoabvon}
\addto\noextrasbulgarian{\autoabvoff}
```

The user can stop babel from turning on the Cyrillic enumeration style by using attribute 'abvoff' when loading babel.

```
\bbl@declare@ttribute{bulgarian}{abvoff}{%
  \PackageInfo{babel}{Turned off automatic Cyrillic enumeration in
    Bulgarian}%
```

```
  \def\autoabvon{\relax}
  \def\autoabvoff{\relax}
  \def\abvon{\enumBul}
}
```

4.8 Cyrillic letters in maths

Set up default Cyrillic math alphabets. To use Cyrillic letters in math mode user should load the `textmath` package *before* loading fontenc package (or babel). Note that by default Cyrillic letters are taken from upright font in math mode (unlike Latin letters).

```
%\RequirePackage{textmath}
\@ifundefined{sym\cyrillicencoding letters}{}{%
\SetSymbolFont{\cyrillicencoding letters}{bold}\cyrillicencoding
  \rmdefault\bfdefault\updefault
\DeclareSymbolFontAlphabet\cyrmathrm{\cyrillicencoding letters}
```

\cyrmathbf
\cyrmathsf
\cyrmathit
\cyrmathtt

And we need a few commands to be able to switch to different variants.

```
\DeclareMathAlphabet\cyrmathbf\cyrillicencoding
  \rmdefault\bfdefault\updefault
\DeclareMathAlphabet\cyrmathsf\cyrillicencoding
  \sfdefault\mddefault\updefault
\DeclareMathAlphabet\cyrmathit\cyrillicencoding
  \rmdefault\mddefault\itdefault
\DeclareMathAlphabet\cyrmathtt\cyrillicencoding
  \ttdefault\mddefault\updefault
```

We define also some bold variants.

```
\SetMathAlphabet\cyrmathsf{bold}\cyrillicencoding
  \sfdefault\bfdefault\updefault
\SetMathAlphabet\cyrmathit{bold}\cyrillicencoding
  \rmdefault\bfdefault\itdefault
}
```

4.9 Alternative names for math functions

Here we define some math operator names in accordance with Bulgarian typesetting traditions.

\tg
\ctg
\cosec
\arctg
\arcctg
\sh
\ch
\cth

Some math functions in Bulgarian have other names, e.g. `sinh` is written as `sh`, etc. We define here alternative math operators for `\tan`, `\cot`, `\csc`, `\arctan`, `\arccot`, `\sinh`, `\cosh`, `\coth`, and `\tanh`.

```
\def\tg{\mathop{\operator@font tg}\nolimits}
\def\ctg{\mathop{\operator@font ctg}\nolimits}
\def\cosec{\mathop{\operator@font cosec}\nolimits}
\def\arctg{\mathop{\operator@font arctg}\nolimits}
\def\arcctg{\mathop{\operator@font arcctg}\nolimits}
\def\sh{\mathop{\operator@font sh}\nolimits}
```

```
\def\ch{\mathop{\operator@font ch}\nolimits}
\def\cth{\mathop{\operator@font cth}\nolimits}
```

\th The macro \th conflicts with \th defined in Latin 1 encoding. We define it as the hyperbolic tangent in math mode but keep the existing definition for text.

```
\addto\extrasbulgarian{%
  \babel@save{\th}%
  \let\ltx@th\th
  \def\th{\textormath{\ltx@th}%
                     {\mathop{\operator@font th}\nolimits}}%
  }
```

\cyrxtounicode The standard font encoding handling for Cyrillic letters uses definitions like \CYRA for the Cyrillic letters. These are not necessary for unicode engines (Luatex, XeTeX) but may still be present in other packages (for example varioref) that handle different languages by storing hard coded strings.

The following command may be used as emergency patch for such problems. It is not to be used routinely as its use may hide unrelated bugs.

```
\def\cyrxtounicode{%
  \let\CYRA=А \let\CYRB=Б \let\CYRV=В \let\CYRG=Г \let\CYRD=Д
  \let\CYRE=Е \let\CYRZH=Ж \let\CYRZ=З \let\CYRI=И \let\CYRISHRT=Й
  \let\CYRK=К \let\CYRL=Л \let\CYRM=М \let\CYRN=Н \let\CYRO=О
  \let\CYRP=П \let\CYRR=Р \let\CYRS=С \let\CYRT=Т \let\CYRU=У
  \let\CYRF=Ф \let\CYRH=Х \let\CYRC=Ц \let\CYRCH=Ч \let\CYRSH=Ш
  \let\CYRSHCH=Щ \let\CYRHRDSN=Ъ \let\CYRSFTSN=Ь \let\CYRYU=Ю
  \let\CYRYA=Я
  \let\cyra=а \let\cyrb=б \let\cyrv=в \let\cyrg=г \let\cyrd=д
  \let\cyre=е \let\cyrzh=ж \let\cyrz=з \let\cyri=и \let\cyrishrt=й
  \let\cyrk=к \let\cyrl=л \let\cyrm=м \let\cyrn=н \let\cyro=о
  \let\cyrp=п \let\cyrr=р \let\cyrs=с \let\cyrt=т \let\cyru=у
  \let\cyrf=ф \let\cyrh=х \let\cyrc=ц \let\cyrch=ч \let\cyrsh=ш
  \let\cyrshch=щ \let\cyrhrdsn=ъ \let\cyrsftsn=ь \let\cyryu=ю
  \let\cyrya=я
}
```

4.10 Compatibility with older versions

This is for compatibility with older Bulgarian packages and support for babel.

```
\DeclareRobustCommand{\No}{%
    \ifmmode{\nfss@text{\textnumero}}\else\textnumero\fi}
```

4.11 Finish

The macro \ldf@finish takes care of looking for a configuration file, setting the main language to be switched on at \begin{document} and resetting the category code of @ to its original value.

```
\ldf@finish{bulgarian}
```

⟨/code⟩

Index

Numbers written in italic refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in roman refer to the code lines where the entry is used.

Symbols

\'' 302
\, 229, 251, 252, 289, 291
\- 323
\@Acdash 328, 332
\@Alph 70, 377, 379, 394–398
\@Alph@bul 348, 394
\@Alph@eng 62, 70, 377, 381, 395, 396
\@Alph@saved 379, 397, 398
\@Bcdash 329, 334
\@Ccdash 330, 336
\@Roman 226
\@SS 304
\@alph 71, 378, 380, 394–398
\@alph@bul 363, 394
\@alph@eng 66, 71, 377, 387, 395, 396
\@alph@saved 380, 397, 398
\@bul@unicodetrue . 8, 9
\@ctrerr 65, 69, 354, 361, 368, 375, 385, 391
\@ifpackageloaded . . 77
\@ifundefined 411
\@nopatterns 4
\@undefined . . 3, 97, 233

\␣ 115, 118, 121, 124, 143, 191, 194, 197, 200, 219, 251

A

\AA 72, 74
\aa 73, 75
\abbgyear 249, 251, 252, 264, 285, 289, 291
\abstractname 104, 154, 180
\abvoff 397, 400
\abvon 397, 399, 408
\adddialect 5
\addto 99, 294, 295, 297, 298, 300, 346, 347, 401, 402, 436
\allowhyphens 316
\alsoname . 142, 171, 218
\appendixname 110, 157, 186
\arcctg 428
\arctg 428
\authorname 123, 162, 199
\autoabvoff 400, 402, 407
\autoabvon 399, 401, 406

B

\babel@save 437
\bbl@activate 299
\bbl@allowhyphens . 323
\bbl@deactivate . . . 301
\bbl@declare@ttribute 403
\bbl@frenchspacing 346
\bbl@nonfrenchspacing 347
\bbl@version . . 97, 233
\begingroup 302
\bfdefault 413, 416, 424, 426
\Bg 54
\bibname . . 106, 155, 182
\Bul 54
\Bulgarian . . . 44, 54–57
\bulgarian@sh@emdash 319, 324
\bulgarian@sh@hyphen 320, 323
\bulgarian@sh@next 319, 322
\bulgarian@sh@tmp 318, 322
\bulgarianhyphenmins 47

C

\captionsbulgarian . 99
\catcode 302
\ccname . . . 134, 167, 210
\cdash 324, 325
\cdp@elt 16
\cdp@list 24
\ch 428
\chaptername 108, 156, 184
\contentsname 112, 158, 188
\cosec 428
\csname 61, 232, 288
\ctg 428
\cth 428
\CurrentOption 344
\cyr 54, 101, 103, 105, 107, 109, 111, 113, 115, 118, 121, 124, 127, 129, 131, 133, 135, 137, 139, 141, 143, 145, 147, 177, 179, 181, 183, 185, 187, 189, 191, 194, 197, 200, 203, 205, 207, 209, 211, 213, 215, 217, 219, 221, 223
\cyrdash 333, 335, 337, 339
\cyrillicencoding 10, 45, 46, 83, 85, 87, 89, 411, 412, 414, 415, 417, 419, 421, 423, 425
\cyrillictext 54, 95, 96, 294
\cyrmathbf 415
\cyrmathit 415, 425
\cyrmathrm 414
\cyrmathsf 415, 423

\cyrmathtt 415
\cyrxtounicode 442

D

\datebulgarian 235
\day 229, 251, 252, 287, 291
\declare@shorthand . 307–314, 317, 343
\DeclareMathAlphabet . 415, 417, 419, 421
\DeclareRobustCommand 44, 49, 458
\DeclareSymbolFontAlphabet 414
\DeclareTextCommand 74, 75
\DeclareTextFontCommand 96
\DeclareTextSymbolDefault 72, 73
\discretionary 315
\dq 302

E

\enclname . 132, 166, 208
\encodingdefault 46, 51
\EndBabelCommands 224, 292
\endcsname . 61, 232, 288
\endgroup 303
\Eng 54
\English 49, 58
\englishhyphenmins . 52
\enumBul . . 394, 397, 408
\enumEng 395
\enumLat 396
\errmessage 331
\exhyphenpenalty . . 335
\expandafter 40, 47, 52, 61, 226, 227, 319, 320
\extrasbulgarian 294, 401, 436

F

\figurename 126, 163, 202
\firstGregorianDatebulgarian 230
\flqq 309
\fontencoding 45, 50, 62, 66
\frqq 310
\futurelet 322

G

\glossaryname 146, 173, 222
\glqq 307
\grqq 308

H

\hbox 312, 341
\headtoname 136, 168, 212
\hskip . 311, 313, 332, 333, 335, 337, 343
\hss 341

I

\if 319
\if@bul@unicode . . 7, 26, 78, 267, 348, 363
\ifcase 62, 66, 236, 349, 356, 364, 370, 382, 388
\ifdefined . . 8, 9, 27, 29
\ifdim 332, 334
\ifmmode 459
\ifx 3, 13, 25, 61, 83, 85, 87, 89, 97, 233, 325, 328–330, 340
\ignorespaces 333, 335, 337, 343
\indexname 120, 161, 196
\initiate@active@char 296
\input 40
\itdefault 420, 426

K

\kern 315

L

\l@bulgarian . . 3, 5, 48
\l@english 53
\language 48, 53
\languageshorthands 297
\lastJulianDatebulgarian 230
\lastskip 332, 334
\lat 95
\latinencoding 10, 50, 51, 62, 66
\latintext 95, 295
\ldf@finish 460
\LdfInit 2
\leavevmode 312, 334, 336
\let 14, 46, 51, 54–58, 70, 71, 95, 377–380, 394–398, 438, 443–456
\listfigurename 114, 159, 190
\listtablename 117, 160, 193
\lowercase 40
\ltx@th 438, 439
\luatexversion . . . 8, 29

M

\mathchar 304
\mathop . . . 428–435, 440
\mddefault 418, 420, 422
\MessageBreak . . . 36, 37
\month . 226, 229, 236, 288
\month@bulgarian 236, 251
\month@Roman 226, 252, 291

N

\newif 7
\nfss@text 459
\No 458
\nobreak 313, 315, 323, 332, 335, 337, 343
\noextrasbulgarian . 295, 300, 347, 402
\nolimits . 428–435, 440
\number . . . 229, 251, 252, 287, 289, 291

O

\operator@font 428–435, 440
\or 62–64, 66–68, 236–247, 349–353, 356–360, 364–368,

370–374,
382–384, 388–390

P

\PackageInfo . . . 42, 98, 150, 234, 255, 404
\PackageWarning 35, 79, 90
\pagename . 138, 169, 214
\partname . 130, 165, 206
\penalty 335
\prefacename 100, 152, 176
\proofname 144, 172, 220
\providehyphenmins 344

R

\r 74, 75
\refname . . 102, 153, 178
\relax . . 40, 61, 406, 407
\RequirePackage . . . 410
\reserved@a 10, 17–23, 82–89, 303, 306
\reserved@b 11, 13
\reserved@c 12–14
\rmdefault 413, 416, 420, 426
\Romannumeral 226
\romannumeral 227, 232, 288

S

\seename . . 140, 170, 216
\selectbglanguage . . 54
\selectenglanguage . 54
\selectfont 45, 50, 62, 66
\selectlanguage . 59, 60
\set@hyphenmins . 47, 52
\SetMathAlphabet 423, 425
\SetString 152–173, 176, 178, 180, 182, 184, 186, 188, 190, 193, 196, 199, 202, 204, 206, 208, 210, 212, 214, 216, 218, 220, 222, 264, 285, 287, 291
\SetStringLoop 257, 261, 269, 277
\SetSymbolFont 412
\sfdefault 418, 424
\sh 428
\space 288
\StartBabelCommands . 151, 175, 256, 265

T

\tablename 128, 164, 204
\tempa@ 327, 328
\tempb@ 327, 329
\tempc@ 327, 330
\tempx@ . . . 326, 328–330
\textcyrillic 96
\textlatin 96
\textnumero 459
\textormath 312, 315, 439
\tg 428
\th 436
\today 251, 287
\todayRoman 229, 252, 291
\ttdefault 422
\tw@ 344

U

\undefined . 25, 325, 340
\unskip 332, 334
\updefault 413, 416, 418, 422, 424
\uppercase 227

W

\weekdaynamebulgarian 230

X

\XeTeXrevision . . . 9, 27

Y

\year 229, 251, 252, 289, 291

Z

\z@ 332, 334
\z@skip . . . 311, 313, 335

Change History

bulgarian-0.99

General: This is a prerelease version of this file. Features needing further testing are removed. . . . 10

bulgarian-1.0b

\extrasbulgarian: Now use \providehyphenmins to provide a default value 21

bulgarian-1.0c

General: Added missing closing brace 12

\dq: repaired typo 19

bulgarian-1.0d

\th: Change definition of \th only for this language. 25

bulgarian-1.0e

\cdash: Two occurences of \emp were changed into tab followed by emp 20

bulgarian-1.1

\captionsbulgarian: Added translation for Glossary. todo: need to check if this is appropriate. 14

Added translation for Proof . . . 14

\latinencoding: Removed. 10
bulgarian-1.1a
General: Commenting out the code in section "Some Greek letters for maths", which is not needed any more. This also resolves a trouble with amsmath, which made it necessary in the past to load the latter before babel. 13
\cyrillicencoding: PackageInfo was commented out, uncommented it 11
\th: Add the local definition to extrasbulgarian, not extrasrussian. 25
bulgarian-1.1b
General: Added EU1 and EU2 and removed LWN 13
Produce warning rather then an error when inputenc has not been loaded 13
\cyrillicencoding: Changing completely the handling of the case when \cyrillicencoding is not defined, see russianb.dtx. The old code is commented out. 11
bulgarian-1.1c
General: Rewrote the text and rearranged the code for the math operators 24
bulgarian-1.1d
\@Alph@bul: New: support for Unicode based engines. 22
\@Alph@eng: Do not define this from scratch 22
\@alph@bul: New: support for Unicode based engines. 22
\@alph@eng: Do not define this from scratch 22
General: Introduced attributes abvon and abvoff 23
Major rewrite of code and documentation for Bulgarian enumerations. 22
\abvoff: New macro. 23
\abvon: New macro. 23
bulgarian-1.2a
General: Switch to babel 3.9g which replaced \UseStrings with a mechanism based on starred \StartBabelCommands commands. 10
bulgarian-1.2b
General: Change the ordinary space between the date and g. to \, in \today and \todayRoman (suggested by Boyko Bantchev). 17
bulgarian-1.2d
General: Removed section „Some Greek letters for maths“. The command had already been commented out in version 1.1a. 13
Substantial clean-up of the code and comments. In particular, removed commented out old code. 10
\cyrillicencoding: Deleted the the old code for the case when the user did not specify font encoding (it had been already commented out in version 1.1b). 11
\extrasbulgarian: Removed the code that made two single quotes automatically produce Bulgarian quotes in Bulgarian text. That code had been commented out for many years, possibly to avoid introducing to additional active characters. 21
bulgarian-1.2e
General: Introduce names of days of week 18
Use \SetStringLoop for months 18
\firstGregorianDatebulgarian: New macro. 17
\lastJulianDatebulgarian: New macro. 17
\weekdaynamebulgarian: New macro. 17
bulgarian-1.2f
General: (bug fix) put \if@bul@unicode before \StartBabelCommands (the dates one) to avoid an error when processing with Luatex. . 18