Babel support for the German language (traditional orthography)

Johannes Braams Bernd Raichle Walter Schmidt
Jürgen Spitzmüller*

v2.10dev (2016/11/05)

Abstract

This manual documents the babel language definition file `germanb.ldf` for German (traditional orthography). The file is part of the babel-german bundle.

1 Aim and usage

The file `germanb.ldf` provides the babel package with all language definition macros (language specific strings and settings) for the German language, including the Austrian and Swiss varieties of German. Furthermore, it assures that the correct hyphenation patterns for the respective language or variety are used.[1] The file adheres to the traditional (1901–1996) orthography. For reformed (post-1996) German orthography support, please refer to the complementary `ngermanb.ldf` file.

In order to use the language definitions provided here, you need to use the babel package and pass the respective language name as an option, either of

- `\usepackage[german]{babel}`
- `\usepackage[austrian]{babel}`
- `\usepackage[swissgerman]{babel}`

Using multiple varieties in parallel is possible; consult the babel manual [2] for details.

2 Shorthands

For all three varieties of German, the character `"` is made active in order to provide some shorthand macros. Some of these shorthands address a peculiarity of traditional German spelling: consonantial character combinations that change in the context of

*Current maintainer. Please report issues via `https://github.com/jspitz/babel-german`.

[1] The file `germanb.ldf` started as a re-implementation of the package `german.sty` (v. 2.5b), which was originally developed by Hubert Partl (cf. [3]) and later maintained by Bernd Raichle (cf. [4]). The re-implementation was done by Johannes Braams.

hyphenations. Other shorthands are provided for frequently used special characters as well as for better control of hyphenation, line breaks and ligatures. Table 1 provides an overview of the shorthands that are provided by `germanb.ldf`.

Table 1: Shorthands provided by `germanb.ldf`

`"a`	Umlaut ⟨ä⟩ (shorthand for `\"a`). Similar shorthands are available for all other lower- and uppercase vowels (umlauts: `"a`, `"o`, `"u`, `"A`, `"O`, `"U`; tremata: `"e`, `"i`, `"E`, `"I`).
`"s`	German ⟨ß⟩ (shorthand for `\ss{}`).
`"z`	German ⟨ß⟩ (shorthand for `\ss{}`). The difference to `"s` is the uppercase version.
`"ck`	⟨ck⟩, hyphenated as ⟨k-k⟩.
`"ff`	⟨ff⟩, hyphenated as ⟨ff-f⟩; this is also implemented for ⟨l⟩, ⟨m⟩, ⟨n⟩, ⟨p⟩, ⟨r⟩ and ⟨t⟩.
`"S`	`\uppercase{"s}`, typeset as ⟨SS⟩ (⟨ß⟩ must be written as ⟨SS⟩ [or ⟨SZ⟩, see below] in uppercase writing).
`"Z`	`\uppercase{"z}`, typeset as ⟨SZ⟩ (⟨ß⟩ must be written as ⟨SZ⟩ [or ⟨SS⟩, see above] in uppercase writing).
`"|`	Disable ligature at this position (e. g., at morpheme boundaries, as in `Auf"|lage`).
`"-`	An additional breakpoint that does still allow for hyphenation at the breakpoints preset in the hyphenation patterns (as opposed to `\-`).
`"=`	An explicit hyphen with a breakpoint, allowing for hyphenation at the other points preset in the hyphenation patterns (as opposed to plain `-`); useful for long compounds such as `IT"=Dienstleisterinnen`.
`"~`	An explicit hyphen without a breakpoint. Useful for cases where the hyphen should stick at the following syllable, e. g., `bergauf und "~ab` .
`""`	A breakpoint that does not output a hyphen if the line break is performed (consider parenthetical extensions as in `(pseudo"~)""wissenschaftlich`).
`"/`	A slash that allows for a linebreak. As opposed to `\slash{}`, hyphenation at the breakpoints preset in the hyphenation patterns is still allowed.
`"‘`	German left double quotes ⟨„⟩.
`"’`	German right double quotes ⟨“⟩.
`"<`	French/Swiss left double quotes ⟨«⟩.
`">`	French/Swiss right double quotes ⟨»⟩.

New feature in v. 2.9!

Table 2 lists some babel macros for quotation marks that might be used as an alternative to the quotation mark shorthands provided by `germanb.ldf`.

Table 2: Alternative commands for quotation marks (provided by babel)

`\glqq`	German left double quotes ⟨„⟩.
`\grqq`	German right double quotes ⟨“⟩.
`\glq`	German left single quotes ⟨‚⟩.
`\grq`	German right single quotes ⟨‘⟩.
`\flqq`	French/Swiss left double quotes ⟨«⟩.
`\frqq`	French/Swiss right double quotes ⟨»⟩.
`\flq`	French/Swiss left single quotes ⟨‹⟩.
`\frq`	French/Swiss right single quotes ⟨›⟩.
`\dq`	The straight quotation mark character ⟨"⟩.

3 Implementation

3.1 General settings

First, we define some helper macros that help us to identify later on which variety of German we are currently dealing with.

```
\def\bbl@opt@german{german}
\def\bbl@opt@germanb{germanb}
\def\bbl@opt@austrian{austrian}
\def\bbl@opt@swissgerman{swissgerman}
```

If `germanb.ldf` is read via the deprecated babel option `germanb`, we make it behave as if `german` was specified.

```
\ifx\CurrentOption\bbl@opt@germanb
  \def\CurrentOption{german}
  \ifx\l@german\@undefined
    \@nopatterns{German (trad. orthography)}
    \adddialect\l@german0
  \fi
  \let\l@germanb\l@german
  \AtBeginDocument{%
    \let\captionsgermanb\captionsgerman
    \let\dategermanb\dategerman
    \let\extrasgermanb\extrasgerman
    \let\noextrasgermanb\noextrasgerman
  }
\fi
```

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

```
\LdfInit\CurrentOption{captions\CurrentOption}
```

If `germanb.ldf` is read as an option, i.e. via `\usepackage` command, `german` could be an 'unknown' language, so we have to make it known. We check for the existence of `\l@german` and issue a warning if it is unknown.

```
\ifx\l@german\@undefined
  \@nopatterns{German (trad. orthography)}
  \adddialect\l@german0
\fi
```

We set `austrian` as a dialect of `german`, since the Austrian variety uses the same hyphenation patterns as Germany's Standard German. If no German patterns are found, we issue a warning.

```
\ifx\CurrentOption\bbl@opt@austrian
  \ifx\l@german\@undefined
    \@nopatterns{German (trad. orthography), needed by Austrian (trad. orthography)}
    \adddialect\l@austrian0
  \else
    \adddialect\l@austrian\l@german
  \fi
\fi
```

For the Swiss variety, we attempt to load the specific `swissgerman` hyphenation patterns and fall back to `german` if those are not available. If no patterns are found, we issue a warning.

```
\ifx\CurrentOption\bbl@opt@swissgerman
  \ifx\l@swissgerman\@undefined
    \ifx\l@german\@undefined
      \@nopatterns{Swiss German (trad. orthography) and German (trad. orthography)}
      \adddialect\l@swissgerman0
    \else
      \@nopatterns{Swiss German (trad. orthography)}
      \adddialect\l@swissgerman\l@german
    \fi
  \fi
\fi
```

3.2 Language-specific strings (captions)

The next step consists of defining macros that provide language specific strings and settings.

`\@captionsgerman` The macro `\@captionsgerman` defines all strings used in the four standard document classes provided with LaTeX for German. This is an internal macro that is inherited and modified by the following macros for the respective language varieties.

```
\@namedef{@captionsgerman}{%
  \def\prefacename{Vorwort}%
  \def\refname{Literatur}%
  \def\abstractname{Zusammenfassung}%
  \def\bibname{Literaturverzeichnis}%
  \def\chaptername{Kapitel}%
  \def\appendixname{Anhang}%
  \def\contentsname{Inhaltsverzeichnis}%
  \def\listfigurename{Abbildungsverzeichnis}%
  \def\listtablename{Tabellenverzeichnis}%
  \def\indexname{Index}%
  \def\figurename{Abbildung}%
  \def\tablename{Tabelle}%
  \def\partname{Teil}%
  \def\enclname{Anlage(n)}%
  \def\ccname{Verteiler}%
  \def\headtoname{An}%
  \def\pagename{Seite}%
  \def\seename{siehe}%
  \def\alsoname{siehe auch}%
  \def\proofname{Beweis}%
  \def\glossaryname{Glossar}%
}
```

`\captionsgerman` The macro `\captionsgerman` is identical to `\@captionsgerman`, but only defined if `german` is requested.

```
\ifx\CurrentOption\bbl@opt@german
  \@namedef{captionsgerman}{%
    \@nameuse{@captionsgerman}%
  }
\fi
```

\captionsaustrian The macro \captionsaustrian builds on \@captionsgerman, but redefines some strings following Austrian conventions (for the respective variants, cf. [1]). It is only defined if austrian is requested.

```
\ifx\CurrentOption\bbl@opt@austrian
  \@namedef{captionsaustrian}{%
    \@nameuse{@captionsgerman}%
    \def\enclname{Beilage(n)}%
  }
\fi
```

\captionsswissgerman The macro \captionsswissgerman builds on \@captionsgerman, but redefines some strings following Swiss conventions (for the respective variants, cf. [1]). It is only defined if swissgerman is requested.

```
\ifx\CurrentOption\bbl@opt@swissgerman
  \@namedef{captionsswissgerman}{%
    \@nameuse{@captionsgerman}%
    \def\enclname{Beilage(n)}%
  }
\fi
```

3.3 Date localizations

\month@german The macro \month@german defines German month names for all varieties.

```
\def\month@german{\ifcase\month\or
  Januar\or Februar\or M\"arz\or April\or Mai\or Juni\or
  Juli\or August\or September\or Oktober\or November\or Dezember\fi}
```

\dategerman The macro \dategerman redefines the command \today to produce German dates. It is only defined if german is requested.

```
\ifx\CurrentOption\bbl@opt@german
  \def\dategerman{\def\today{\number\day.~\month@german
      \space\number\year}}
\fi
```

\dateswissgerman The macro \dateswissgerman does the same for Swiss German dates. It is only defined if swissgerman is requested. The result is identical to German.

```
\ifx\CurrentOption\bbl@opt@swissgerman
  \def\dateswissgerman{\def\today{\number\day.~\month@german
      \space\number\year}}
\fi
```

\dateaustrian The macro \dateaustrian redefines the command \today to produce Austrian versions of the German dates. Here, the naming of January („Jänner“) differs from the other German varieties. The macro is only defined if austrian is requested.

```
\ifx\CurrentOption\bbl@opt@austrian
  \def\dateaustrian{\def\today{\number\day.~\ifnum1=\month
    J\"anner\else \month@german\fi \space\number\year}}
\fi
```

3.4 Extras

\extrasgerman
\extrasaustrian
\extrasswissgerman
\noextrasaustrian
\noextrasswissgerman
\noextrasgerman

The macros \extrasgerman, \extrasaustrian and \extrasswissgerman, respectively, will perform all the extra definitions needed for the German language or the respective variety. The macro \noextrasgerman is used to cancel the actions of \extrasgerman. \noextrasaustrian and \noextrasswissgerman behave analoguously.

First, the character " is declared active for all German varieties. This is done once, later on its definition may vary.

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

Depending on the option with which the language definition file has been loaded, the macro \extrasgerman, \extrasaustrian or \extrasswissgerman is defined. Each of those is identical: they load the shorthands defined below and activate the " character.

```
\@namedef{extras\CurrentOption}{%
  \languageshorthands{german}}
\expandafter\addto\csname extras\CurrentOption\endcsname{%
  \bbl@activate{"}}
```

Next, again depending on the option with which the language definition file has been loaded, the macro \noextrasgerman, \noextrasaustrian or \noextrasswissgerman is defined. These deactivate the " character and thus turn the shorthands off again outside of the respective variety.

```
\expandafter\addto\csname noextras\CurrentOption\endcsname{%
  \bbl@deactivate{"}}
```

In order for TEX to be able to hyphenate German words which contain ‘ß’ (in the OT1 position ^^Y) we have to give the character a nonzero \lccode (see Appendix H, the TEXbook).

```
\expandafter\addto\csname extras\CurrentOption\endcsname{%
  \babel@savevariable{\lccode25}%
  \lccode25=25}
```

The umlaut accent macro \" is changed to lower the umlaut dots. The redefinition is done with the help of \umlautlow.

```
\expandafter\addto\csname extras\CurrentOption\endcsname{%
  \babel@save\"\umlautlow}

\expandafter\addto\csname noextras\CurrentOption\endcsname{%
  \umlauthigh}
```

The German hyphenation patterns can be used with \lefthyphenmin and \righthyphenmin set to 2.

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

For German texts we need to assure that `\frenchspacing` is turned on.

```
\expandafter\addto\csname extras\CurrentOption\endcsname{%
  \bbl@frenchspacing}
\expandafter\addto\csname noextras\CurrentOption\endcsname{%
  \bbl@nonfrenchspacing}
```

3.5 Active characters, macros & shorthands

The following code is necessary because we need an extra active character. This character is then used as indicated in table 1.

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

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

```
\begingroup \catcode`\"12
\def\x{\endgroup
  \def\@SS{\mathchar"7019 }
  \def\dq{"}}
\x
```

Now we can define the doublequote shorthands: the umlauts,

```
\declare@shorthand{german}{"a}{\textormath{\"{a}\allowhyphens}{\ddot a}}
\declare@shorthand{german}{"o}{\textormath{\"{o}\allowhyphens}{\ddot o}}
\declare@shorthand{german}{"u}{\textormath{\"{u}\allowhyphens}{\ddot u}}
\declare@shorthand{german}{"A}{\textormath{\"{A}\allowhyphens}{\ddot A}}
\declare@shorthand{german}{"O}{\textormath{\"{O}\allowhyphens}{\ddot O}}
\declare@shorthand{german}{"U}{\textormath{\"{U}\allowhyphens}{\ddot U}}
```

tremata,

```
\declare@shorthand{german}{"e}{\textormath{\"{e}}{\ddot e}}
\declare@shorthand{german}{"E}{\textormath{\"{E}}{\ddot E}}
\declare@shorthand{german}{"i}{\textormath{\"{\i}}%
                               {\ddot\imath}}
\declare@shorthand{german}{"I}{\textormath{\"{I}}{\ddot I}}
```

German ß,

```
\declare@shorthand{german}{"s}{\textormath{\ss}{\@SS{}}}
\declare@shorthand{german}{"S}{\SS}
\declare@shorthand{german}{"z}{\textormath{\ss}{\@SS{}}}
\declare@shorthand{german}{"Z}{SZ}
```

German and French/Swiss quotation marks,

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

discretionary commands

```
\declare@shorthand{german}{"c}{\textormath{\bbl@disc ck}{c}}
\declare@shorthand{german}{"C}{\textormath{\bbl@disc CK}{C}}
\declare@shorthand{german}{"F}{\textormath{\bbl@disc F{FF}}{F}}
\declare@shorthand{german}{"l}{\textormath{\bbl@disc l{ll}}{l}}
\declare@shorthand{german}{"L}{\textormath{\bbl@disc L{LL}}{L}}
\declare@shorthand{german}{"m}{\textormath{\bbl@disc m{mm}}{m}}
\declare@shorthand{german}{"M}{\textormath{\bbl@disc M{MM}}{M}}
\declare@shorthand{german}{"n}{\textormath{\bbl@disc n{nn}}{n}}
\declare@shorthand{german}{"N}{\textormath{\bbl@disc N{NN}}{N}}
\declare@shorthand{german}{"p}{\textormath{\bbl@disc p{pp}}{p}}
\declare@shorthand{german}{"P}{\textormath{\bbl@disc P{PP}}{P}}
\declare@shorthand{german}{"r}{\textormath{\bbl@disc r{rr}}{r}}
\declare@shorthand{german}{"R}{\textormath{\bbl@disc R{RR}}{R}}
\declare@shorthand{german}{"t}{\textormath{\bbl@disc t{tt}}{t}}
\declare@shorthand{german}{"T}{\textormath{\bbl@disc T{TT}}{T}}
```

(we need to treat `"f` a bit differently in order to preserve the ff-ligature)

```
\declare@shorthand{german}{"f}{\textormath{\bbl@discff}{f}}
\def\bbl@discff{\penalty\@M
  \afterassignment\bbl@insertff \let\bbl@nextff= }
\def\bbl@insertff{%
  \if f\bbl@nextff
    \expandafter\@firstoftwo\else\expandafter\@secondoftwo\fi
  {\relax\discretionary{ff-}{f}{ff}\allowhyphens}{f\bbl@nextff}}
\let\bbl@nextff=f
```

and some additional commands (hyphenation, line breaking and ligature control):

```
\declare@shorthand{german}{"-}{\nobreak\-\bbl@allowhyphens}
\declare@shorthand{german}{"|}{%
  \textormath{\penalty\@M\discretionary{-}{}{\kern.03em}%
              \allowhyphens}{}}
\declare@shorthand{german}{""}{\hskip\z@skip}
\declare@shorthand{german}{"~}{\textormath{\leavevmode\hbox{-}}{-}}
\declare@shorthand{german}{"=}{\penalty\@M-\hskip\z@skip}
\declare@shorthand{german}{"/}{\textormath
  {\bbl@allowhyphens\discretionary{/}{}{/}\bbl@allowhyphens}{}}
```

`\mdqon` `\mdqoff` `\ck`

All that's left to do now is to define a couple of commands for reasons of compatibility with `german.sty`.

```
\def\mdqon{\shorthandon{"}}
\def\mdqoff{\shorthandoff{"}}
\def\ck{\allowhyphens\discretionary{k-}{k}{ck}\allowhyphens}
```

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\CurrentOption
```

3.6 `austrian.ldf`, `german.ldf` and `swissgerman.ldf`

Babel expects a ⟨*lang*⟩`.ldf` file for each ⟨*lang*⟩. So we create portmanteau ldf files for `austrian`, `german` and `swissgerman`.[2] These files themselves only load `germanb.ldf`, which does the real work:

177 `\input germanb.ldf\relax`

Change History

[2] For `austrian` and `german`, this is not strictly necessary, since babel provides aliases for these languages (pointing to `germanb`). However, since babel does not officially support these aliases anymore after the language definition files have been separated from the core, we provide the whole range of ldf files for the sake of completeness.

References

[1] Ammon, Ulrich et al.: *Variantenwörterbuch des Deutschen. Die Standardsprache in Österreich, der Schweiz und Deutschland sowie in Liechtenstein, Luxemburg, Ostbelgien und Südtirol.* Berlin, New York: De Gruyter.

[2] Braams, Johannes and Bezos, Javier: *Babel.* http://www.ctan.org/pkg/babel.

[3] Partl, Hubert: *German TEX*, *TUGboat* 9/1 (1988), p. 70–72.

[4] Raichle, Bernd: *German.* http://www.ctan.org/pkg/german.