

The bicaption package*

Axel Sommerfeldt

<https://gitlab.com/axelsommerfeldt/caption>

2023/03/11

Abstract

This package supports the typesetting of bilingual captions.

Contents

1	Loading the package	2
2	Setting options	2
3	Additional options	3
4	The <code>\bicaption</code> commands	5
5	Customising lists	5
6	Language Selection	7
7	Required packages	10
8	Supported packages	10
8.1	Support of the listings package	11
8.2	Support of the longtable package	12
8.3	Support of the subcaption package	12
9	The implementation	16
9.1	Identification	16
9.2	Initial code	16
9.3	Declaration of options	17
9.4	Execution of options	19
9.5	Main code	20
9.6	Support of the listings package	24
9.7	Support of the longtable package	25

*This package has version number v1.6.

1 Loading the package

This package will be loaded by

```
\usepackage[\langle options \rangle]{bicaption} .
```

The options for the bicaption package are the same ones as for the caption package and specify settings which are used for the second language *additionally*. In fact

```
\usepackage[\langle options \rangle]{bicaption}
```

is identical to

```
\usepackage{bicaption}  
\bicaptionsetup{\langle options \rangle} .
```

When used with the babel or polyglossia package, the bicaption package should be loaded *after* it, so the main language will be set automatically. See [section 6](#) for details.

2 Setting options

`\bicaptionsetup` The `\bicaptionsetup` command sets options specifically for bilingual captions.

New feature
v1.6

```
\bicaptionsetup{\langle options for 1st language \rangle}{\langle options for 2nd language \rangle}
```

sets options which will be used for the first resp. second heading of the bilingual captions *additionally* to the ones which are setup for the specific floating environment.

To limit bilingual options to specific environments one can use an optional argument for `\bicaptionsetup`, e.g.:

```
\bicaptionsetup[figure]{...}{...}
```

will limit the settings to the bilingual headings of `figure` environments only.

Options specified with `\usepackage[...]{bicaption}` and `\bicaptionsetup{...}{...}` will override the ones specified by `\captionsetup{...}` and `\captionsetup[figure]{...}` (same for 'table'). So finally we have the following order how settings for bilingual captions are applied:

1. Global settings (`\usepackage[...]{caption}` and `\captionsetup{...}`)
2. Environmental settings (`\captionsetup[figure -or- table]{...}`)
3. Local settings (`\captionsetup{...}` inside `figure` or `table` environment)
4. Custom bilingual settings (`\usepackage[...]{bicaption}` and `\bicaptionsetup{...}{...}`)
5. Environmental bilingual settings (`\bicaptionsetup[figure -or- table]{...}{...}`)

An example:

```

\usepackage[labelsep=quad,indentation=10pt]{caption}
\usepackage[labelfont=bf]{biccaption}
\captionsetup[table]{labelfont=it,position=top}

```

causes the second heading of the bilingual caption inside `table` environments to be typeset with the settings

```
labelsep=quad,indentation=10pt,position=top,labelfont=bf.
```

▲ Internally the `\biccaptionsetup` uses the `\captionsetup` command, i.e.

```
\biccaptionsetup{\langle options for 1st language \rangle}{\langle options for 2nd language \rangle}
```

is identical to

```

\captionsetup[bi-first]{\langle options for 1st language \rangle}
\captionsetup[bi-second]{\langle options for 2nd language \rangle}

```

and

```
\biccaptionsetup[figure]{...}{...}
```

is identical to

```

\captionsetup[figure][bi-first]{...}
\captionsetup[figure][bi-second]{...}

```

Prior to v1.6 of this package this was the way options had to be specified. This still works (and will continue to work in the future), but is not recommended, `\biccaptionsetup` should be used instead.

3 Additional options

These options are available additional to the ones offered by the `caption` package:

`lang=` Sets the language of the caption, e.g.

```
\usepackage[lang=english]{biccaption}
```

will typeset the second caption of bilingual captions in English. (The language will be set with `\selectcaptionlanguage` internally, see [section 6](#) for details.)

`bi-lang=` Causes a selection of the headings of bilingual captions.

```
\captionsetup{bi-lang=both}
```

will cause that both caption headings are being typeset. (This is the default.)

```
\captionsetup{bi-lang=first}
```

will cause that only the *first* heading is being typeset, and

```
\captionsetup{bi-lang=second}
```

will cause that only the *second* heading is being typeset.

```
bi-slc= \captionsetup{bi-slc=bool}
```

switches the common single-line-check on or off, i.e. when switched on only a single check will be done for both captions, and the result will affect both captions afterwards. So if only one caption is longer than a single line, both captions will be treated as if they are longer than a single line, even if the second one isn't. (The default is on.)

```
bi-swap= \captionsetup{bi-swap}
```

will swap the primary and secondary language, making the first language the second one and vice versa. (The default is false.)

New feature
v1.3

```
bi-separator=
```

The vertical distance between the first and second bilingual caption is usually determined by the \TeX skips set by `\normalsize` which is applied right before both captions get typesetted. (This behaviour is inherited from the original \LaTeX code for `\caption` which applies `\normalsize` right before `\@makecaption`, too.)

```
\captionsetup{bi-separator=name}
```

will select an additional separator between first and second bilingual caption. You could choose one of the following: 'none' (which is the default one and could also be addressed as 'default'), 'smallskip', 'medskip', 'largeskip', or a self-defined one using

```
\DeclareBiCaption-  
Separator
```

```
\DeclareBiCaptionSeparator{name}{code}
```

Examples:

```
\captionsetup{bi-separator=smallskip}
```

will put a `\smallskip` between the two bilingual captions.

```
\DeclareBiCaptionSeparator{hrule}{\hrule}  
\captionsetup{bi-separator=hrule}
```

will draw a horizontal line between the two bilingual captions.

```
\DeclareBiCaptionSeparator{3pt}{\vspace{3pt}}  
\captionsetup{bi-separator=3pt}
```

will place 3pt extra vertical space between the two bilingual captions.

Note: In contrast to the original \LaTeX code for `\caption` the caption package does not apply `\normalsize` directly but will apply the caption font definition `normalsize` instead (which is usually defined as `\normalsize`). Therefore the vertical space between both captions could also be influenced by redefining it, e.g.: `\DeclareCaptionFont{normalsize}{...}`

Important: All options starting with 'bi-' must be applied using `\captionsetup` and *NOT* using `\bicaptionsetup`. This is because they do not alter the setting of the caption for the 1st or 2nd language specifically, but instead alter the behaviour how bilingual captions are set in general.

4 The `\bication` commands

`\bication` Bilingual captions will be typeset by

```
\bication[ $\langle list\ entry\ \#1 \rangle$ ]{ $\langle heading\ \#1 \rangle$ }
      [ $\langle list\ entry\ \#2 \rangle$ ]{ $\langle heading\ \#2 \rangle$ }
\bication*{ $\langle heading\ \#1 \rangle$ }{ $\langle heading\ \#2 \rangle$ }
```

The `\label` should be placed either after this command, or inside the first heading.

`\bicationbox` Bilingual caption boxes will be typeset by

```
\bicationbox[ $\langle list\ entry\ \#1 \rangle$ ]{ $\langle heading\ \#1 \rangle$ }
      [ $\langle list\ entry\ \#2 \rangle$ ]{ $\langle heading\ \#2 \rangle$ }
      [ $\langle width \rangle$ ] [ $\langle inner-pos \rangle$ ]{ $\langle contents \rangle$ }
\bicationbox*{ $\langle heading\ \#1 \rangle$ }{ $\langle heading\ \#2 \rangle$ }
      [ $\langle width \rangle$ ] [ $\langle inner-pos \rangle$ ]{ $\langle contents \rangle$ }
```

The `\label` should be placed inside the first heading.

(For a description of the optional parameters $\langle width \rangle$ and $\langle inner-pos \rangle$ please take a look at the caption package documentation, `\captionbox`.)

5 Customising lists

`list=` As default both caption texts will be insert into the List of Figures resp. List of Tables. To suppress the second entry just pass the option `list=off` to the `bication` package, e.g.:

```
\usepackage[lang=english,...,list=off]{bication}
```

or

```
\usepackage[...] {bication}
...
\bicationsetup{}{list=off}
```

`listtype+=` Another option is separating the lists. For that purpose the option

```
listtype+= $\langle list\ type\ extension \rangle$ 
```

can be used to tell the `bication` package to use a different list for the second caption text. The given value will be appended to the current environment type; for example with `listtype+=X` the list entries will be put into the list responsible for the types `figureX` (= `figure + X`), `tableX` (= `table + X`) etc.

Such a $\langle list\ type \rangle$ can be defined using `\DeclareFloatingEnvironment` offered by the `newfloat` package, but some document classes or other packages offer macros for defining new floating environment types (and their corresponding lists) as well.

A sample document:

```

\documentclass[a4paper]{article}

% Use "ngerman" as 1st language, "english" as 2nd one
\usepackage[english,ngerman]{babel}

% Load the bicaption package with 2nd language set to
% "english", and list type "figureEng" resp. "tableEng"
\usepackage[lang=english,listtype+=Eng]{bicaption}

\usepackage{newfloat}
% Define type "figureEng" and \listoffigureEng
\DeclareFloatingEnvironment[fileext=lof2]{figureEng}
    [Figure][List of Figures]
% Define type "tableEng" and \listoftableEng
\DeclareFloatingEnvironment[fileext=lot2]{tableEng}
    [Table][List of Tables]

\begin{document}

\listoffigures    % typeset "Abbildungsverzeichnis"
\listoffigureEng % typeset "List of Figures"

\begin{figure}
  \centering
  A placeholder for an image or whatever
  \bicaption{Deutscher Text}{English text}
\end{figure}

\end{document}

```

A different approach is using one list for both languages, but with different formatting. Since the caption package does not offer options and commands for customising the format of the lists, one needs an additional package for this purpose, for example the `titletoc` package:

```

\documentclass[a4paper]{article}

% Use "ngerman" as 1st language, "english" as 2nd one
\usepackage[english,ngerman]{babel}

% Load the bicaption package with 2nd language set to
% "english", and list type "figure2" resp. "table2"
\usepackage[lang=english,listtype+=2]{bicaption}

% We load the titletoc package for customising lists
% Note: Loading titletoc should be done prior
% defining additional floating environments with
% \DeclareFloatingEnvironment
\usepackage{titletoc}

```

```

\usepackage{newfloat}
% Define the new floating environment type "figure2"
% Use the same file extension as for "figure" (.lof) here
\DeclareFloatingEnvironment[fileext=lof]{figure2}
% Define the new floating environment type "table2"
% Use the same file extension as for "table" (.lot) here
\DeclareFloatingEnvironment[fileext=lot]{table2}

% We use the titletoc package for customising "figure2"
% which is appropriate for the second language captions
\titlecontents{figure2}[3.8em]
  {} % no above code
  {} % empty numbered entry format
  {} % empty numberless entry format
  {} % empty filler page format

\begin{document}

\renewcommand\listfigurename
  {Abbildungsverzeichnis / List of Figures}
\listoffigures

\begin{figure}
  \centering
  A placeholder for an image or whatever
  \bicaption{Deutscher Text}{English text}
\end{figure}

\end{document}

```

6 Language Selection

For language selection the bicaption package uses two macros internally:

`\captionmainlanguage` `\captionmainlanguage` contains the main language, for example ‘french’ or ‘german’. If not set manually, the bicaption package will try to obtain this setting from the babel or polyglossia package after the preamble of the document, i.e. at `\begin{document}`.

So if you are using either babel or polyglossia, and want to inherit the main language setting from it, then simply forget about the `\captionmainlanguage` stuff and skip the rest of the section.

Otherwise one can define `\captionmainlanguage` manually, e.g.:

```

\newcommand\captionmainlanguage{french}
\usepackage[lang=english]{bicaption}

```

Note: Prior to v1.5 `\captionmainlanguage` needed to be defined *before* loading the bicaption package. Since v1.5 it could be defined either before or after loading the bicaption package.

`\selectcaptionlanguage` `\selectcaptionlanguage` will be used internally to select the language:

New feature
v1.1

```
\selectcaptionlanguage{<font-or-list-entry>}{<language>}
```

For setting the language of the caption *<font-or-list-entry>* will be `\@firstoftwo`, for setting the language of the list entry *<font-or-list-entry>* will be `\@secondoftwo`.¹ It defaults to `\select@language` (caption) resp. `\selectlanguage` (list entry) offered by the `babel` and `polyglossia` package:

```
\providecommand*\selectcaptionlanguage[2]{%  
  #1{\select@language}{\selectlanguage}{#2}}
```

If you need to alter this, just either define `\selectcaptionlanguage` prior loading the `bicaption` package, or redefine it afterwards.

Example document using `babel`:

```
\documentclass[a4paper]{article}  
  
% Use "ngerman" as 1st language, "english" as 2nd one  
\usepackage[english,ngerman]{babel}  
  
% Add custom translations to babel  
\addto\captionsgerman{%  
% \renewcommand\whatevername{Wasauchimmer}%  
% ...  
%}  
\addto\captionsenglish{%  
% \renewcommand\whatevername{Whatever}%  
% ...  
%}  
  
% Load the bicaption package with 2nd language set to  
% "english"  
\usepackage[lang=english]{bicaption}  
  
\begin{document}  
  
\begin{figure}  
  \centering  
  A placeholder for an image or whatever  
  \bicaption{Deutscher Text}{English text}  
\end{figure}  
  
\end{document}
```

The same example document but using a custom implementation of `\captionmainlanguage` and `\selectcaptionlanguage` instead of `babel`:

¹`\@firstoftwo` and `\@secondoftwo` are defined in the \LaTeX kernel and simply pick either the 1st or 2nd argument.


```

\documentclass[a4paper]{article}

% Load the bicaption package with 2nd language set to
% "english"
\usepackage[lang=english]{bicaption}

% Set "german" as main bi-caption language
\newcommand\captionmainlanguage{german}

% Declare an own language switching mechanism
% for bi-captions (instead of using babel)
\renewcommand\selectcaptionlanguage[2]{%
  \csname captions#2\endcsname}
\newcommand\captionsgerman{%
  \renewcommand\figurename{Abbildung}%
  \renewcommand\tablename{Tabelle}%
  % \renewcommand\whatevertime{Wasauchimmer}%
  % ...
}
\newcommand\captionsenglish{%
  \renewcommand\figurename{Figure}%
  \renewcommand\tablename{Table}%
  % \renewcommand\whatevertime{Whatever}%
  % ...
}

\begin{document}

\begin{figure}
  \centering
  A placeholder for an image or whatever
  \bicaption{Deutscher Text}{English text}
\end{figure}

\end{document}

```

New feature v1.5 Since v1.5 a warning is issued if the main language could neither be detected automatically nor was it set explicitly by the user. If you really don't want to set languages for bi-captions but are annoyed by the warning you could trick the bicaption by defining a custom dummy language-selection mechanism, e.g.:

```

\newcommand\captionmainlanguage{dummy}
\renewcommand\selectcaptionlanguage[2]{}

```

\DeclareCaptionLangOption For internal implementation reasons the selection of language will be done delayed, i.e. not done immediately at `lang=<language>`. So if you do

New feature v1.2 `\bicaptionsetup}{lang=ngerman, labelsep=quad}`

the language `ngerman` will only be stored internally, and the label separator will be set to `quad` afterwards. Some time later, right before the caption is actually typeset, the language will be set to `ngerman`.

Usually this is no problem, but think of options which will be overwritten by the language selection, or options which act on the language currently set, for example

```
\bicaptionsetup{}{lang=ngerman,name=Bild} .
```

`lang=ngerman` changes the environment name to “Abbildung”, and `name=Bild` changes the environment name to “Bild”. One would expect that the name is finally “Bild”, but because of the delayed nature of `lang=ngerman` it will be “Abbildung” instead, at least if we don’t take action about this.

For that reason the command

```
\DeclareCaptionLangOption{<caption option name>}
```

is offered. Options handled this way will be applied twice if used after the `lang=` option, when the option is actually used, and right after the language is selected.

```
\DeclareCaptionLangOption{name}
```

will be done by the `bicaption` package automatically, since the environment name will usually be overwritten by a language selection. So actually

```
\bicaptionsetup{}{lang=ngerman,name=Bild}
```

will give the expected result, i.e. the environment name is typeset as “Bild”.

7 Required packages

New feature Starting with version 1.4 the `bicaption` package requires at least version 3.2 of the `caption` package and loads it automatically. (Older versions of the `bicaption` package have required exactly the version of the `caption` package which was released with it.)

v1.4

If you need to use a specific version of the `caption` package you need to load it *before* the `bicaption` package, e.g.:

```
\usepackage[...]{caption}[=v3.5]
\usepackage[...]{bicaption}
```

Note that there are limitations if an older version of the `caption` package is used:

- Full support of list entries of the `lstlisting` environment (offered by the `listings` package) needs at least `caption` v3.6.

8 Supported packages

The `bicaption` package was adapted to the following packages which deals with captions, too: `listings` [2], `longtable` [3], and `subcaption` [4].

8.1 Support of the listings package

New feature v1.5 If the listings package [2] is loaded, the listings options `caption1` and `caption2` are available additionally, where option `caption1` specifies the caption of the first language and `caption2` of the second one.

Example document, using distinctive lists for each language:

```
\documentclass[a4paper]{article}
\usepackage{graphicx}

% Use "ngerman" as 1st language, "english" as 2nd one
\usepackage[english,ngerman]{babel}

% Load the bicaption package with 2nd language set to
% "english", and list type "figureEng" resp. "tableEng"
\usepackage[lang=english,listtype+=Eng,font=it]{bicaption}
\captionsetup{slc=off} % do not center short captions

\usepackage{listings}

% Set German names
\addto\captionsgerman{%
  \renewcommand\lstlistingname{Quelltext}%
  \renewcommand\lstlistlistingname{Quelltextverzeichnis}%
}
\AtBeginDocument{\captionsgerman}% or load listings before babel

% Set English names
\addto\captionsenglish{%
  \renewcommand\lstlistingname{Listing}%
  \renewcommand\lstlistlistingname{List of Listings}%
}

\usepackage{newfloat}
% Define the new floating environment type "lstlistingEng"
% (just to get an extra list for English listing captions)
\DeclareFloatingEnvironment[fileext=lol2]{lstlistingEng}
    [Listing][List of Listings]

\begin{document}

\lstlistoflistings % German
\listoflstlistingEng % English

\clearpage

\begin{lstlisting}
[language=C,
caption1=Deutscher Titel,
caption2=English Title]
```

```

int main()
{
    printf( "Hello world!\n" );
    return 0;
}
\end{lstlisting}

\end{document}

```

8.2 Support of the longtable package

If the longtable package [3] is loaded, \bicaption is available in the longtable environment as well, e.g.:

```

\documentclass[a4paper]{article}

% Use "ngerman" as 1st language, "english" as 2nd one
\usepackage[english,ngerman]{babel}

% Load the bicaption package with 2nd language set to
% "english"
\usepackage[lang=english]{bicaption}

\usepackage{longtable}

\begin{document}

\begin{longtable}{ll}
    \bicaption{Deutscher Titel}{English Title}\\
    A & B \\
    C & D \\
    ...
\end{longtable}

\end{document}

```

8.3 Support of the subcaption package

If the subcaption package [4] is loaded, these commands are available additionally:

`\bisubcaption` Bilingual sub-captions will be typeset by

```

\bisubcaption[\langle list entry #1 \rangle]{\langle heading #1 \rangle}
             [\langle list entry #2 \rangle]{\langle heading #2 \rangle}
\bisubcaption*{\langle heading #1 \rangle}{\langle heading #2 \rangle}

```

The `\label` should be placed either after this command, or inside the first heading.

`\bisubcaptionbox` Bilingual sub-caption boxes will be typeset by

```

\bisubcaptionbox[⟨list entry #1⟩]{⟨heading #1⟩}
    [⟨list entry #2⟩]{⟨heading #2⟩}
    [⟨width⟩][⟨inner-pos⟩]{⟨contents⟩}
\bisubcaptionbox*{⟨heading #1⟩}{⟨heading #2⟩}
    [⟨width⟩][⟨inner-pos⟩]{⟨contents⟩}

```

The `\label` should be placed inside the first heading.

(For a description of the optional parameters `⟨width⟩` and `⟨inner-pos⟩` please take a look at the [subcaption package documentation](#), `\subcaptionbox`.)

8.3.1 A sample document

```

\documentclass[english,ngerman]{article}
\usepackage{selinput}
\SelectInputMappings{adieresis={ä},germandbls={ß}}

\usepackage{babel}
\usepackage[lang=english,font=it]{bicaption}
\usepackage[format=hang]{subcaption}

\begin{document}

\begin{figure}[!htb]
  \centering
  \bisubcaptionbox
    {Teilabbildung A\label{fig:test:A}}
    {Subfigure A}[0.4\textwidth]{IMAGE}%
  \quad
  \bisubcaptionbox
    {Teilabbildung langer Titel B\label{fig:test:B}}
    {Subfigure long title B}[0.4\textwidth]{IMAGE}%
  \bicaption{Deutscher Titel}{English Title}
  \label{fig:test}
\end{figure}

\captionsetup{bi-lang=both}

\begin{figure}[!htb]
  \centering
  \bisubcaptionbox[A]
    {Und eine gaaaanz lange Caption: Teilabbildung A}
    {Subfigure A}[0.4\textwidth]{IMAGE}%
  \quad
  \bisubcaptionbox[B]
    {Teilabbildung B}
    {Subfigure B}[0.4\textwidth]{IMAGE}%
  \bicaption[Abbildungsverzeichnistitel]
    {Und eine noch viel viel viel
    längere deutsche Beschriftung: Deutscher Titel}

```

```

        {Short English heading}
\end{figure}

\captionsetup{bi-slc=off}

\begin{figure}[!htb]
  \centering
  \bisubcaptionbox[A]
    {Und eine gaaaanz lange Caption: Teilabbildung A}
    {Subfigure A}[0.4\textwidth]{IMAGE}%
  \quad
  \bisubcaptionbox[B]
    {Teilabbildung B}
    {Subfigure B}[0.4\textwidth]{IMAGE}%
  \bicaption[Abbildungsverzeichnistitel]
    {Und eine noch viel viel viel
    längere deutsche Beschriftung: Deutscher Titel}
    {Short English heading}
\end{figure}

\captionsetup{slc=off}

\begin{figure}[!htb]
  \centering
  \bisubcaptionbox[A]
    {Und eine gaaaanz lange Caption: Teilabbildung A}
    {Subfigure A}[0.4\textwidth]{IMAGE}%
  \quad
  \bisubcaptionbox[B]
    {Teilabbildung B}
    {Subfigure B}[0.4\textwidth]{IMAGE}%
  \bicaption[Abbildungsverzeichnistitel]
    {Und eine noch viel viel viel
    längere deutsche Beschriftung: Deutscher Titel}
    {Short English heading}
\end{figure}

\end{document}

```

IMAGE
 (a) Teilabbildung A
(a) Subfigure A

IMAGE
 (b) Teilabbildung langer Titel B
(b) Subfigure long title B

Abbildung 1: Deutscher Titel
Figure 1: English Title

IMAGE	IMAGE
(a) Und eine gaaaanz lange Caption: Teilabbildung A <i>(a) Subfigure A</i>	(b) Teilabbildung B <i>(b) Subfigure B</i>

Abbildung 2: Und eine noch viel viel viel längere deutsche Beschriftung: Deutscher Titel
Figure 2: Short English heading

IMAGE	IMAGE
(a) Und eine gaaaanz lange Caption: Teilabbildung A <i>(a) Subfigure A</i>	(b) Teilabbildung B <i>(b) Subfigure B</i>

Abbildung 3: Und eine noch viel viel viel längere deutsche Beschriftung: Deutscher Titel
Figure 3: Short English heading

IMAGE	IMAGE
(a) Und eine gaaaanz lange Caption: Teilabbildung A <i>(a) Subfigure A</i>	(b) Teilabbildung B <i>(b) Subfigure B</i>

Abbildung 4: Und eine noch viel viel viel längere deutsche Beschriftung: Deutscher Titel
Figure 4: Short English heading

9 The implementation

9.1 Identification

We need at least L^AT_EX₂ ϵ version 1994/12/01.

```
1 \NeedsTeXFormat{LaTeX2e}[1994/12/01]
```

Bypass the release declarations in case the L^AT_EX kernel doesn't know how to deal with them (as suggested by <https://www.latex-project.org/publications/2018-FMi-TUB-tbl22mitt-version-rollback.pdf>).

```
2 \providecommand\DeclareRelease[3]{}
3 \providecommand\DeclareCurrentRelease[2]{}
4 \DeclareCurrentRelease{v1}{2011/07/13}
```

Declare all supported releases.

```
4 \DeclareCurrentRelease{v1}{2011/07/13}
```

Identify the current version of the package.

```
5 \ProvidesPackage{biccaption}[2023/02/19 v1.6-dev Bilingual Captions (AR)]
```

Since we base on the caption package we load it here.

```
6 \RequirePackage{caption}[2011/11/10] % we need at least v3.2e
```

9.2 Initial code

```
\biccaption@Info \biccaption@Info{<message>}
7 \newcommand*\biccaption@Info[1]{%
8 \PackageInfo{biccaption}{#1}}
```

```
\biccaption@InfoNoLine \biccaption@InfoNoLine{<message>}
Note: The \@gobble at the end of the 2nd argument of \PackageInfo suppresses the line
number info. See TLC2[5], A.4.7, p885 for details.
```

```
9 \newcommand*\biccaption@InfoNoLine[1]{%
10 \PackageInfo{biccaption}{#1\@gobble}}
```

```
\biccaption@Warning \biccaption@Warning{<message>}
11 \newcommand*\biccaption@Warning[1]{%
12 \biccaption@WarningNoLine{#1\on@line}}
```

```
\biccaption@WarningNoLine \biccaption@WarningNoLine{<message>}
13 \newcommand*\biccaption@WarningNoLine[1]{%
14 \PackageWarning{biccaption}{#1.^J\biccaption@wh\@gobbletwo}}
15 \newcommand*\biccaption@wh{%
16 See the 'biccaption' package documentation for explanation.}
```

```
\biccaption@Error \biccaption@Error{<message>}
17 \newcommand*\biccaption@Error[1]{%
18 \PackageError{biccaption}{#1}{\caption@eh{biccaption}}}
19 \providecommand*\caption@@eh[1]{%
20 If you do not understand this error, please take a closer look\MessageBreak
21 at the documentation of the '#1' package, especially the\MessageBreak
22 section about errors.\MessageBreak\@ehc}
```


9.3 Declaration of options

The option `bi-lang` will setup which language(s) will actually be typeset, the first one, the second one, or both of them.

```
23 \newcount\bi-caption@lang
24 \DeclareCaptionOption{bi-lang}{%
25   \caption@ifinlist{#1}{0,all,both}{%
26     \bi-caption@lang=0\relax
27   }{\caption@ifinlist{#1}{1,1st,first}{%
28     \bi-caption@lang=1\relax
29   }{\caption@ifinlist{#1}{2,2nd,second}{%
30     \bi-caption@lang=2\relax
31   }}{%
32     \bi-caption@Error{Undefined bi-lang value `#1'}%
33   }}}
```

The option `bi-singlelinecheck` will setup if a single check will be used for both languages (`=on`), or if both languages will be checked individually (`=off`).

```
34 \DeclareCaptionOption{bi-singlelinecheck}[1]{%
35   \caption@set@bool\bi-caption@ifslc{#1}}
36 \DeclareCaptionOption{bi-slc}[1]{%
37   \caption@set@bool\bi-caption@ifslc{#1}}
```

The option `bi-swap` will swap the primary and secondary language, making the first language the second one and vice versa.

```
38 \DeclareCaptionOption{bi-swap}[1]{%
39   \caption@set@bool\bi-caption@ifswap{#1}}
```

The option `bi-separator` will setup the separator between the two bi-lingual captions.

```
40 \DeclareCaptionOption{bi-separator}[1]{%
41   \caption@set{biseparator}{#1}}
42 \providecommand*\caption@set[2]{%
43   \@ifundefined{caption@#1@#2}%
44     {\caption@Error{Undefined #1 `#2'}}%
45     {\expandafter\let\csname caption@#1\expandafter\endcsname\csname caption@#1@#2
```

```
\DeclareBiCaptionSeparator \DeclareBiCaptionSeparator{<name>}{<code>}
```

```
46 \newcommand\DeclareBiCaptionSeparator[2]{%
47   \caption@decl{biseparator}{#1}{#2}}
48 \@onlypreamble\DeclareBiCaptionSeparator
49 \providecommand*\caption@decl[2]{%
50   \global\long\expandafter\def\csname caption@#1@#2\endcsname}
```

There are four pre-defined separators, called ‘none’, ‘smallskip’, ‘medskip’, and ‘bigskip’.

```
51 \DeclareBiCaptionSeparator{none}{}
52 \DeclareBiCaptionSeparator{smallskip}{\smallskip}
53 \DeclareBiCaptionSeparator{medskip}{\medskip}
54 \DeclareBiCaptionSeparator{bigskip}{\bigskip}
```

The default separator usually maps to ‘none’.

```
55 \caption@ifundefined\SetCaptionDefault
56   {\def\caption@biseparator@default{\caption@biseparator@none}}
57   {\SetCaptionDefault{biseparator}{none}}
```

The option `lang=<language>` will setup the language of the caption. We can't set the language immediately because otherwise we will get in trouble, e.g. when using the `microtype` package via `\usepackage [babel] {microtype}`. So we store the selected language to `\bication@language` instead and will set it later on. But this has a drawback, option settings which will be overwritten by the selection of the language do not work correctly anymore. Therefore we will save specific options and set them (again) after setting the language.

```
58 \DeclareCaptionOption{lang}{%
59   \caption@ifundefined\bication@language
60     \bication@language@setupkeys
61     \relax
62   \edef\bication@language{#1}%
63   \let\bication@language@setoptions\@empty
64   \global\let\bication@language@value\bication@language}
```

Map `language=` to `lang=`.

```
65 \let\KV@caption@language\KV@caption@lang
```

```
\DeclareCaptionLangOption \DeclareCaptionLangOption{<option key>}
```

Saves the gives `<option key>` to the list `\bication@language@keylist`. These options will be stored and set (again) after setting the language.

```
66 \newcommand*\bication@language@keylist{}
67 \newcommand*\DeclareCaptionLangOption[1]{%
68   \@ifundefined{KV@caption@#1}%
69     {\bication@Error{Undefined caption option `#1'}}%
70     {\@cons\bication@language@keylist{{#1}}}}
71 \let\DeclareCaptionLanguageOption\DeclareCaptionLangOption
72 \newcommand*\bication@language@setupkeys{%
73   \def\@elt##1{%
74     \expandafter\let\csname KV@bication@##1\expandafter\endcsname
75       \csname KV@caption@##1\endcsname
76     \@namedef{KV@caption@##1}{\bication@KV{##1}}}%
77   \bication@language@keylist
78   \let\@elt\relax}
79 \newcommand\bication@KV[2]{%
80   \@bication@KV{#1}{#2}%
81   \l@addto@macro\bication@language@setoptions{%
82     \@bication@KV{#1}{#2}}
83 \newcommand\@bication@KV[1]{%
84   \@nameuse{KV@bication@#1}}
```

The only option affected I'm aware of is the `name=` option, so this will be saved and set later on.

```
85 \DeclareCaptionLangOption{name}
```

```
\bication@selectlanguage \bication@selectlanguage{<font-or-list-entry>}
```

Set the language (stored in `\bication@language`) via `\selectbication-language`.

```
86 \newcommand*\bication@selectlanguage[1]{%
87   \caption@ifundefined\bication@language{}{%
88     \expandafter\selectcaptionlanguage\expandafter#1\expandafter
```

```

89     {\bication@language}%
90     \bication@language@setoptions}}
\caption@beginhook (of the caption package kernel) will be extended here so the
language setting will actually take effect.
91 \g@addto@macro\caption@beginhook{%
92   \bication@selectlanguage\@firstoftwo}

```

`\selectcaptionlanguage` `\selectcaptionlanguage{⟨font-or-list-entry⟩}{⟨language⟩}` will finally be used to set up the language.

```

93 \providecommand*\selectcaptionlanguage[1]{%
94   #1\select@language\selectlanguage}

```

9.4 Execution of options

Setup default values for `bi-lang`, `bi-singlelinecheck`, `bi-swap`, and `bi-separator`.

```

95 \caption@ExecuteOptions{caption}{bi-lang=0,bi-slc=1,bi-swap=0,bi-separator=default}

```

Set the language for the first caption. (Since *v1.5a* we do this `\AtBeginDocument` so `\captionmainlanguage` could be defined *after* loading the `bication` package resp. `babel` or `polyglossia` could be loaded *after* the `bication` package.)

```

96 \AtBeginDocument{%
97   \ifcsname captionmainlanguage\endcsname
98     \bication@InfoNoLine{%
99       Main language is set to \captionmainlanguage}%
100  \else
101    \bication@setmainlanguage
102  \fi

103  \ifcsname captionmainlanguage\endcsname
104    \captionsetup*[bi-first]{lang=\captionmainlanguage}%
105  \else
106    \ifcsname bication@language@value\endcsname % option "lang=" was used
107      \expandafter\bication@Warning
108    \else
109      \expandafter\bication@Info
110    \fi
111    {Main language is not set}%
112  \fi}

```

`\bication@setmainlanguage` `\bication@setmainlanguage` is used to define `\captionmainlanguage` if not already done.

```

113 \newcommand*\bication@setmainlanguage{%
114   \ifcsname bbl@main@language\endcsname
115     \bication@InfoNoLine{%
116       babel found, main language is \bbl@main@language}%
117     \let\captionmainlanguage\bbl@main@language
118   \else\ifcsname xpg@main@language\endcsname
119     \bication@InfoNoLine{%
120       polyglossia found, main language is \xpg@main@language}%
121     \let\captionmainlanguage\xpg@main@language
122   \fi\fi}
123 \@onlypreamble\bication@setmainlanguage

```

We use `\caption@ProcessOptions` here to add the options to the ‘bi-second’ option list instead of executing them immediately.

```
124 \caption@SetupOptions{bicaption}{\captionsetup*[bi-second]{#2}}%
125 \caption@ProcessOptions*{bicaption}
```

9.5 Main code

`\caption@addcontentsline` We patch `\caption@addcontentsline` (of the caption package kernel) so `bicaption@addcontentsline` will be used for bilingual captions instead.

```
126 \let\bicaption@addcontentsline@ORI\caption@addcontentsline
127 \renewcommand*\caption@addcontentsline[2]{%
128   \bicaption@LT@setup
129   \global\let\bicaption@LT@setup\relax
130   \caption@ifundefined\bicaption@lentry
131     {\bicaption@addcontentsline@ORI{#1}{#2}}%
132     {\expandafter\bicaption@addcontentsline\expandafter
133       {\bicaption@lentry}{#1}{#2}}%
134     \global\let\bicaption@lentry\@undefined}}
```

`\bicaption@addcontentsline` `\bicaption@addcontentsline{<list entry #2>}{<type>}{<list entry #1>}`
Typeset both captions using the original version of `\caption@addcontentsline`.

```
135 \newcommand\bicaption@addcontentsline[3]{%
136   \begingroup
137   \let\caption@addsubcontentslines\@gobble
138   \caption@setoptions{bi}%
Execute the options setup with \captionsetup[bi]{...}.
```

Do the first list entry, if requested.

```
139   \ifnum\bicaption@lang=2\relax \else
140     \begingroup
141       \caption@setoptions{bi-first}%
142       \bicaption@@addcontentsline{#2}{#3}%
143     \endgroup
144   \fi
```

Do the second list entry, if requested.

```
145   \ifnum\bicaption@lang=1\relax \else
146     \begingroup
147       \caption@setoptions{bi-second}%
148       \bicaption@@addcontentsline{#2}{#1}%
149     \endgroup
150   \fi
```

```
151 \endgroup
152 \caption@addsubcontentslines{#2}}
153 \newcommand*\bicaption@@addcontentsline[2]{%
154   \caption@ifcontentsline{#2}{%
155     \bicaption@selectlanguage\@secondoftwo
156     \bicaption@addcontentsline@ORI{#1}{#2}}}
```

`\caption@make` We redefine `\caption@make` (of the caption package kernel) so `\bicaption@make` will be used for bilingual captions instead.

```

157 \caption@ifundefined\caption@make@
158   {\let\caption@make@\caption@make}
159   {}

160 \renewcommand\caption@make[2]{%
161   \caption@ifundefined\bicaption@text
162     {\begingroup
163       \caption@make#{1}#{2}%
164     \endgroup}%
165   {\expandafter\bicaption@make\expandafter
166     {\bicaption@text}#{1}#{2}%
167   \global\let\bicaption@text\undefined}%
168   \@nameuse{caption@make@epilogue}}

```

`\bicaption@make` `\bicaption@make{<text #2>}{<label>}{<text #1>}`

Typeset both captions using the original version of `\caption@make`.

```

169 \newcommand\bicaption@make[3]{%

```

Execute the options setup with `\captionsetup[bi]{...}`.

```

170   \caption@setoptions{bi}%

```

Perform the common single-line-check for both captions, if requested.

```

171   \ifnum\bicaption@lang=0\relax
172     \bicaption@ifslc
173       {\bicaption@slc{#2}{#3}}{\caption@setsinglelinecheck0}%
174       \bicaption@slc{#2}{#1}}{\caption@setsinglelinecheck0}}%
175     {}%
176   \fi

```

Typeset the first caption, if requested. (Otherwise we only apply the label of it.)

```

177   \ifnum\bicaption@lang=2\relax
178     \caption@thelabel
179     \let\bicaption@tempa\relax
180   \else
181     \begingroup
182       \caption@setoptions{bi-first}%
183       \caption@make{#2}{#3}%
184     \endgroup
185     \def\bicaption@tempa{\bicaption@separator\caption@biseparator}%
186   \fi

```

Typeset the second caption, if requested.

```

187   \ifnum\bicaption@lang=1\relax
188   \else
189     \bicaption@tempa
190     \begingroup
191       \caption@setoptions{bi-second}%
192       \caption@make{#2}{#1}%
193     \endgroup
194   \fi

```

```

195   \global\bicaption@clrlabel}

```

```

196 \newcommand*\bicaption@separator{\par}

```

```

197 \caption@ifundefined\caption@@slc
198   {\newcommand\bicaption@slc{\caption@slc}}
199   {\newcommand\bicaption@slc[2]{\caption@@slc{#1}{#2}{\captionwidth}}}
200 \providecommand*\caption@setsinglelinecheck{%
201   \caption@set@bool\caption@ifslc}

```

9.5.1 The `\bicaption` commands

`\bicaptionsetup` `\bicaptionsetup* [⟨type⟩] {⟨keyval-list of options 1⟩} {⟨keyval-list of options 2⟩}`
applies the given list of options.

```

202 \newcommand*\bicaptionsetup{%
203   \caption@teststar\@bicaptionsetup*{} }
204 \newcommand*\@bicaptionsetup[1]{%
205   \kernel@ifnextchar[%
206     {\bicaption@setup@options{#1}}%
207     {\bicaption@setup*} }
208 \def\bicaption@setup@options#1[#2]{%
209   \bicaption@setup{#1[#{2}]}}
210 \newcommand*\bicaption@setup[2]{%
211   \captionsetup#1[bi-first]{#2}%
212   \captionsetup#1[bi-second]}

```

`\bicaption` `\bicaption* [⟨list entry #1⟩] {⟨text #1⟩} [⟨list entry #2⟩] {⟨text #2⟩}`
213 `\newcommand\bicaption{\@bicaption\caption}`

`\bicaptionbox` `\bicaptionbox* [⟨entry #1⟩] {⟨text #1⟩} [⟨entry #2⟩] {⟨text #2⟩} [⟨...⟩] {⟨...⟩}`
214 `\newcommand\bicaptionbox{\@bicaption\captionbox}`

`\bisubcaption` `\bisubcaption* [⟨list entry #1⟩] {⟨text #1⟩} [⟨list entry #2⟩] {⟨text #2⟩}`
215 `\newcommand\bisubcaption{\@bicaption\subcaption}`
216 `\let\subbicaption\bisubcaption`

`\bisubcaptionbox` `\bisubcaptionbox* [⟨entry #1⟩] {⟨text #1⟩} [⟨entry #2⟩] {⟨text #2⟩} [⟨...⟩] {⟨...⟩}`
217 `\newcommand\bisubcaptionbox{\@bicaption\subcaptionbox}`
218 `\let\subbicaptionbox\bisubcaptionbox`

`\@bicaption` `\@bicaption{⟨cmd⟩} * [⟨entry #1⟩] {⟨text #1⟩} [⟨entry #2⟩] {⟨text #2⟩} ...`
219 `\newcommand*\@bicaption[1]{%`
220 `\@ifstar`
221 `{\def\bicaption@cmd{\bicaption@star{#1}}%`
222 `\@@bicaption}%`
223 `{\def\bicaption@cmd{#1}%`
224 `\caption@dblarg\@@@bicaption}}`
225 `\newcommand\@@@bicaption[1]{%`
226 `\@@@@bicaption{#1}[]}`
227 `\long\def\@@@bicaption[#1]#2{%`
228 `\caption@dblarg{\@@@@bicaption{#1}{#2}}}`

```

229 \long\def\@@@bicaption#1#2[#3]#4{%
230   \bicaption@getlabel{#2}%
231   \bicaption@ifswap
232     {\bicaption@set{#1}{#2}%
233     \bicaption@cmd[#{3}]{#4}}%
234     {\bicaption@set{#3}{#4}%
235     \bicaption@cmd[#{1}]{#2}}
236 \long\def\bicaption@star#1[#2]{#1*}

```

```
\bicaption@getlabel \bicaption@getlabel{<text>}
```

gets the label command out of the given caption text and stores it to `\caption@thelabel`. It uses `\caption@getlabel` for this purpose which interface unfortunately has changed over time. (Changing the interface was a bad idea in the first place but now it is as it is).

```

237 \@ifundefined{caption@getlabel}{%
238   \newcommand\bicaption@getlabel[1]{%   caption3 < v1.7
239     \bicaption@getlabel#1\label{} \@nil}
240   \long\def\bicaption@getlabel#1\label#2#3\@nil{%
241     \def\@tempa{#2}%
242     \ifx\@tempa\@empty
243       \let\caption@thelabel\relax
244     \else
245       \def\@tempb{*}%
246       \ifx\@tempa\@tempb
247         \def\caption@thelabel{\label*}%
248         \bicaption@@@getlabel#3\@nil
249       \else
250         \def\caption@thelabel{\label}%
251         \bicaption@@@getlabel{#2}#3\@nil
252       \fi
253     \fi}
254   \long\def\bicaption@@@getlabel#1#2\@nil{%
255     \def\@tempa{#1}%
256     \def\@tempb{[]}%
257     \ifx\@tempa\@tempb
258       \bicaption@@@getlabel#1#2\@nil
259     \else
260       \l@addto@macro\caption@thelabel{#{#1}}%
261     \fi}
262   \long\def\bicaption@@@getlabel[#1]#2\@nil{%
263     \l@addto@macro\caption@thelabel{[#{#1}]}%
264     \bicaption@@@getlabel#2\@nil}
265 }{\@ifundefined{caption@@@getlabel}{%
266   \newcommand\bicaption@getlabel[1]{%   caption3 >= v1.7
267     \caption@getlabel#1\label{} \@nil}
268 }{%
269   \newcommand*\bicaption@getlabel{%   caption3 >= v2.0
270     \caption@getlabel}
271 }}

```

```

\bicaption@clrlabel \bicaption@clrlabel
resets \caption@thelabel to \relax.
272 \@ifundefined{caption@clrlabel}{%
273   \newcommand*\bicaption@clrlabel{\let\caption@thelabel\relax} % caption3 < v2.3
274 }{%
275   \newcommand*\bicaption@clrlabel{\caption@clrlabel} % caption3 >= v2.3
276 }

\bicaption@set \bicaption@set{<list-entry>}{<text>}
initiates the bilingual caption typesetting by storing the extra texts into \bicaption@l-
entry and \bicaption@text.
277 \newcommand\bicaption@set[2]{%
278   \def\bicaption@lentry{#1}%
279   \def\bicaption@text{\ignorespaces #2}}

\bicaption@clear \bicaption@clear
clears the stuff stored by \bicaption@set.
280 \newcommand*\bicaption@clear{%
281   \let\bicaption@lentry\@undefined
282   \let\bicaption@text\@undefined}

\caption@freeze To make \bicaption work inside SCfigure and FPfigure environments we need
to add \bicaption to \caption@freeze.
283 \AtBeginDocument{%
284   \ifx\caption@freeze\@undefined \else
285     \g@addto@macro\caption@freeze{%
286       \let\caption@frozen@bicaption\bicaption
287       \def\bicaption{%
288         \caption@withoptargs\caption@SC@bicaption}%
289       \long\def\caption@SC@bicaption#1#2{%
290         \kernel@ifnextchar[%]
291           {\caption@SC@bi@caption{#1}{#2}}%
292           {\caption@SC@bi@caption@{#1}{#2}}}%
293       \long\def\caption@SC@bi@caption#1#2[#3]#4{%
294         \caption@@freeze{\bicaption#1{#2}[#3]}{#4}}%
295         \ignorespaces}%
296       \long\def\caption@SC@bi@caption@#1#2#3{%
297         \caption@@freeze{\bicaption#1{#2}{#3}}%
298         \ignorespaces}%
299       \l@addto@macro\caption@warmup{%
300         \let\bicaption\caption@frozen@bicaption}}%
301     \fi}

```

9.6 Support of the listings package

```

302 \caption@IfPackageLoaded{listings}[2004/02/13 v1.2]{%
If the listings package is loaded, we define the new options 'caption1' and 'caption2'.
303   \lst@Key{caption1}\relax{%
304     \lstKV@OptArg[#1]{#1}{%
305       \bicaption@ifswap

```



```

306         {\bication@set{##1}{##2}}%
307         {\def\lst@@caption{##1}\def\lst@caption{##2}}}%
308     \let\lst@title\@empty}%
309 \lst@Key{caption2}\relax{%
310     \lstKV@OptArg[{}]{#1}{#1}{%
311         \bication@ifswap
312         {\def\lst@@caption{##1}\def\lst@caption{##2}}}%
313         {\bication@set{##1}{##2}}}%
314     \let\lst@title\@empty}%
315 \lst@AddToHookExe{TextStyle}{%
316     \bication@clear}%
317 }{}

```

9.7 Support of the longtable package

`\LT@bication` Same as `\@bication` but for longtable (offered by the longtable package). `\bication@LTsetup` will be executed later on, inside `\LT@makecaption` offered by the caption package.

```

318 \newcommand\LT@bication{%
319     \noalign\bgroup
320     \@ifstar
321     {\gdef\bication@cmd{\LT@c@ption\@gobble}%
322     \LT@@@bication}%
323     {\gdef\bication@cmd{\LT@c@ption\@firstofone}%
324     \caption@dblarg\LT@@@bication}}
325 \newcommand\LT@@@bication[1]{%
326     \LT@@@@bication}{#1}[]
327 \long\def\LT@@@@bication[#1]#2{%
328     \caption@dblarg{\LT@@@@bication{#1}{#2}}}
329 \long\def\LT@@@@@bication#1#2[#3]#4{%
330     \gdef\bication@LTsetup{%
331         \bication@getlabel{#2}%
332         \bication@LT@setup}%
333     \gdef\bication@LT@setup{%
334         \bication@ifswap
335         {\bication@set{#1}{#2}}%
336         {\bication@set{#3}{#4}}}%
337     \bication@ifswap
338     {\egroup\bication@cmd[{}]{#4}}%
339     {\egroup\bication@cmd[{}]{#2}}}
340 \let\bication@LTsetup\relax
341 \let\bication@LT@setup\relax

```

`\caption@LT@setup` Execute the stuff defined by `\LT@bication` to prepare the typesetting of the longtable bilingual caption.

```

342 \caption@ifundefined\caption@LT@setup{%
343     % Fallback code for caption v3.2
344     \captionsetup*[longtable]{bication-lt-setup}
345     \DeclareCaptionOptionNoValue{bication-lt-setup}{%
346         \bication@LTsetup
347         \global\let\bication@LTsetup\relax}
348 }%

```

```
349 \g@addto@macro\caption@LT@setup{%  
350   \bicaption@LTsetup  
351   \global\let\bicaption@LTsetup\relax}  
352 }
```

References

- [1] Axel Sommerfeldt:
Customising captions of floating environments,
2022/01/07
- [2] Carsten Heinz & Brooks Moses:
The Listings Package,
2007/02/22
- [3] David Carlisle:
The longtable package,
2004/02/01
- [4] Axel Sommerfeldt:
The subcaption package,
2022/01/07
- [5] Frank Mittelbach and Michel Goossens:
The L^AT_EX Companion (2nd. Ed.),
Addison-Wesley, 2004.