The refcount package

Heiko Oberdiek*

2019/12/15 v3.6

Abstract

References are not numbers, however they often store numerical data such as section or page numbers. `\ref` or `\pageref` cannot be used for counter assignments or calculations because they are not expandable, generate warnings, or can even be links. The package provides expandable macros to extract the data from references. Packages hyperref, nameref, titleref, and babel are supported.

Contents

*Please report any issues at https://github.com/ho-tex/refcount/issues

1 Usage

1.1 Setting counters

The following commands are similar to LaTeX's `\setcounter` and `\addtocounter`, but they extract the number value from a reference:

`\setcounterref`, `\addtocounterref`
`\setcounterpageref`, `\addtocounterpageref`

They take two arguments:

`\...counter...ref {`⟨*LaTeX counter*⟩`} {`⟨*reference*⟩`}`

An undefined references produces the usual LaTeX warning and its value is assumed to be zero. Example:

```
\newcounter{ctrA}
\newcounter{ctrB}
\refstepcounter{ctrA}\label{ref:A}
\setcounterref{ctrB}{ref:A}
\addtocounterpageref{ctrB}{ref:A}
```

1.2 Expandable commands

These commands that can be used in expandible contexts (inside calculations, `\edef`, `\csname`, `\write`, ...):

`\getrefnumber`, `\getpagerefnumber`

They take one argument, the reference:

`\get...refnumber {`⟨*reference*⟩`}`

The default for undefined references can be changed with macro `\setrefcountdefault`, for example this package calls:

```
\setrefcountdefault{0}
```

Since version 2.0 of this package there is a new command:

`\getrefbykeydefault {`⟨*reference*⟩`} {`⟨*key*⟩`} {`⟨*default*⟩`}`

This generalized version allows the extraction of further properties of a reference than the two standard ones. Thus the following properties are supported, if they are available:

Key	Description	Package
⟨*empty*⟩	same as `\ref`	LaTeX
`page`	same as `\pageref`	LaTeX
`title`	section and caption titles	titleref
`name`	section and caption titles	nameref
`anchor`	anchor name	hyperref
`url`	url/file	hyperref/xr

Since version 3.2 the expandable macros described before in this section are expandable in exact two expansion steps.

1.3 Undefined references

Because warnings and assignments cannot be used in expandible contexts, undefined references do not produce a warning, their values are assumed to be zero. Example:

```
\label{ref:here}% somewhere
\refused{ref:here}% see below
\ifodd\getpagerefnumber{ref:here}%
  reference is on an odd page
\else
  reference is on an even page
\fi
```

In case of undefined references the user usually want's to be informed. Also LaTeX prints a warning at the end of the LaTeX run. To notify LaTeX and get a normal warning, just use

`\refused {`⟨*reference*⟩`}`

outside the expanding context. Example, see above.

1.3.1 Check for undefined references

In version 3.2 macros were added, that test, whether references are defined.

`\IfRefUndefinedExpandable` {⟨*refname*⟩} {⟨*then*⟩} {⟨*else*⟩}
`\IfRefUndefinedBabel` {⟨*refname*⟩} {⟨*then*⟩} {⟨*else*⟩}

If the reference is not available and therefore undefined, then argument ⟨*then*⟩ is executed, otherwise argument ⟨*else*⟩ is called. Macro `\IfRefUndefinedExpandable` is expandable, but ⟨*refname*⟩ must not contain babel shorthand characters. Macro `\IfRefUndefinedBabel` supports shorthand characters of babel, but it is not expandable.

1.4 Notes

- The method of extracting the number in this package also works in cases, where the reference cannot be used directly, because a package such as hyperref has added extra stuff (hyper link), so that the reference cannot be used as number any more.
- If the reference does not contain a number, assignments to a counter will fail of course.

2 Implementation

```
1 ⟨*package⟩
```

Reload check, especially if the package is not used with LaTeX.

```
2 \begingroup\catcode61\catcode48\catcode32=10\relax%
3   \catcode13=5 % ^^M
4   \endlinechar=13 %
5   \catcode35=6 % #
6   \catcode39=12 % '
7   \catcode44=12 % ,
8   \catcode45=12 % -
9   \catcode46=12 % .
10  \catcode58=12 % :
11  \catcode64=11 % @
12  \catcode123=1 % {
13  \catcode125=2 % }
14  \expandafter\let\expandafter\x\csname ver@refcount.sty\endcsname
15  \ifx\x\relax % plain-TeX, first loading
16  \else
17    \def\empty{}%
18    \ifx\x\empty % LaTeX, first loading,
19      % variable is initialized, but \ProvidesPackage not yet seen
20    \else
21      \expandafter\ifx\csname PackageInfo\endcsname\relax
22        \def\x#1#2{%
23          \immediate\write-1{Package #1 Info: #2.}%
24        }%
25      \else
26        \def\x#1#2{\PackageInfo{#1}{#2, stopped}}%
27      \fi
28      \x{refcount}{The package is already loaded}%
29      \aftergroup\endinput
30    \fi
31  \fi
32 \endgroup%
```

Package identification:

```
33 \begingroup\catcode61\catcode48\catcode32=10\relax%
34   \catcode13=5 % ^^M
35   \endlinechar=13 %
36   \catcode35=6 % #
37   \catcode39=12 % '
38   \catcode40=12 % (
39   \catcode41=12 % )
40   \catcode44=12 % ,
41   \catcode45=12 % -
42   \catcode46=12 % .
43   \catcode47=12 % /
44   \catcode58=12 % :
45   \catcode64=11 % @
46   \catcode91=12 % [
47   \catcode93=12 % ]
48   \catcode123=1 % {
49   \catcode125=2 % }
50   \expandafter\ifx\csname ProvidesPackage\endcsname\relax
51     \def\x#1#2#3[#4]{\endgroup
52       \immediate\write-1{Package: #3 #4}%
53       \xdef#1{#4}%
54     }%
```

```
  \else
    \def\x#1#2[#3]{\endgroup
      #2[{#3}]%
      \ifx#1\@undefined
        \xdef#1{#3}%
      \fi
      \ifx#1\relax
        \xdef#1{#3}%
      \fi
    }%
  \fi
\expandafter\x\csname ver@refcount.sty\endcsname
\ProvidesPackage{refcount}%
  [2019/12/15 v3.6 Data extraction from label references (HO)]%
\begingroup\catcode61\catcode48\catcode32=10\relax%
  \catcode13=5 % ^^M
  \endlinechar=13 %
  \catcode123=1 % {
  \catcode125=2 % }
  \catcode64=11 % @
  \def\x{\endgroup
    \expandafter\edef\csname rc@AtEnd\endcsname{%
      \endlinechar=\the\endlinechar\relax
      \catcode13=\the\catcode13\relax
      \catcode32=\the\catcode32\relax
      \catcode35=\the\catcode35\relax
      \catcode61=\the\catcode61\relax
      \catcode64=\the\catcode64\relax
      \catcode123=\the\catcode123\relax
      \catcode125=\the\catcode125\relax
    }%
  }%
\x\catcode61\catcode48\catcode32=10\relax%
\catcode13=5 % ^^M
\endlinechar=13 %
\catcode35=6 % #
\catcode64=11 % @
\catcode123=1 % {
\catcode125=2 % }
\def\TMP@EnsureCode#1#2{%
  \edef\rc@AtEnd{%
    \rc@AtEnd
    \catcode#1=\the\catcode#1\relax
  }%
  \catcode#1=#2\relax
}
\TMP@EnsureCode{33}{12}% !
\TMP@EnsureCode{39}{12}% '
\TMP@EnsureCode{42}{12}% *
\TMP@EnsureCode{45}{12}% -
\TMP@EnsureCode{46}{12}% .
\TMP@EnsureCode{47}{12}% /
\TMP@EnsureCode{91}{12}% [
\TMP@EnsureCode{93}{12}% ]
\TMP@EnsureCode{96}{12}% `
\edef\rc@AtEnd{\rc@AtEnd\noexpand\endinput}
```

2.1 Loading packages

```
\begingroup\expandafter\expandafter\expandafter\endgroup
\expandafter\ifx\csname RequirePackage\endcsname\relax
  \input ltxcmds.sty\relax
  \input infwarerr.sty\relax
\else
  \RequirePackage{ltxcmds}[2011/11/09]%
  \RequirePackage{infwarerr}[2010/04/08]%
\fi
```

2.2 Defining commands

\rc@IfDefinable

```
\ltx@IfUndefined{@ifdefinable}{%
  \def\rc@IfDefinable#1{%
    \ifx#1\ltx@undefined
      \expandafter\ltx@firstofone
    \else
      \ifx#1\relax
        \expandafter\expandafter\expandafter\ltx@firstofone
      \else
        \@PackageError{refcount}{%
          Command \string#1 is already defined.\MessageBreak
          It will not redefined by this package%
        }\@ehc
        \expandafter\expandafter\expandafter\ltx@gobble
      \fi
    \fi
  }%
}{%
  \let\rc@IfDefinable\@ifdefinable
}
```

\rc@RobustDefOne
\rc@RobustDefZero

```
\ltx@IfUndefined{protected}{%
  \ltx@IfUndefined{DeclareRobustCommand}{%
    \def\rc@RobustDefOne#1#2#3#4{%
      \rc@IfDefinable#3{%
        #1\def#3##1{#4}%
      }%
    }%
    \def\rc@RobustDefZero#1#2{%
      \rc@IfDefinable#1{%
        \def#1{#2}%
      }%
    }%
  }{%
    \def\rc@RobustDefOne#1#2#3#4{%
      \rc@IfDefinable#3{%
        \DeclareRobustCommand#2#3[1]{#4}%
      }%
    }%
    \def\rc@RobustDefZero#1#2{%
      \rc@IfDefinable#1{%
        \DeclareRobustCommand#1{#2}%
      }%
    }%
  }%
}{%
  \def\rc@RobustDefOne#1#2#3#4{%
```

```
    \rc@IfDefinable#3{%
      \protected#1\def#3##1{#4}%
    }%
  }%
  \def\rc@RobustDefZero#1#2{%
    \rc@IfDefinable#1{%
      \protected\def#1{#2}%
    }%
  }%
}
```

\rc@newcommand

```
\ltx@IfUndefined{newcommand}{%
  \def\rc@newcommand*#1[#2]#3{% hash-ok
    \rc@IfDefinable#1{%
      \ifcase#2 %
        \def#1{#3}%
      \or
        \def#1##1{#3}%
      \or
        \def#1##1##2{#3}%
      \else
        \rc@InternalError
      \fi
    }%
  }%
}{%
  \let\rc@newcommand\newcommand
}
```

2.3 \setrefcountdefault

\setrefcountdefault

```
\rc@RobustDefOne\long{}\setrefcountdefault{%
  \def\rc@default{#1}%
}

\setrefcountdefault{0}
```

2.4 \refused

\refused

```
\ltx@IfUndefined{G@refundefinedtrue}{%
  \rc@RobustDefOne{}{*}\refused{%
    \begingroup
      \csname @safe@activestrue\endcsname
      \ltx@IfUndefined{r@#1}{%
        \protect\G@refundefinedtrue
        \rc@WarningUndefined{#1}%
      }{}%
    \endgroup
  }%
}{%
  \rc@RobustDefOne{}{*}\refused{%
    \begingroup
      \csname @safe@activestrue\endcsname
      \ltx@IfUndefined{r@#1}{%
```

```
        \csname protect\expandafter\endcsname
        \csname G@refundefinedtrue\endcsname
        \rc@WarningUndefined{#1}%
      }{}%
    \endgroup
  }%
}
```

`\rc@WarningUndefined`

```
\ltx@IfUndefined{@latex@warning}{%
  \def\rc@WarningUndefined#1{%
    \ltx@ifundefined{thepage}{%
      \def\thepage{\number\count0 }%
    }{}%
    \@PackageWarning{refcount}{%
      Reference `#1' on page \thepage\space undefined%
    }%
  }%
}{%
  \def\rc@WarningUndefined#1{%
    \@latex@warning{%
      Reference `#1' on page \thepage\space undefined%
    }%
  }%
}
```

2.5 Setting counters by reference data

2.5.1 Generic setting

`\rc@set` Generic command for `\{set,addto}counter{page,}ref`:

- #1: `\setcounter`, `\addtocounter`
- #2: `\ltx@car` (for `\ref`), `\ltx@cartwo` (for `\pageref`)
- #3: LaTeX counter
- #4: reference

```
\def\rc@set#1#2#3#4{%
  \begingroup
    \csname @safe@activestrue\endcsname
    \refused{#4}%
    \expandafter\rc@@set\csname r@#4\endcsname{#1}{#2}{#3}%
  \endgroup
}
```

`\rc@@set`

- #1: `\r@<...>`
- #2: `\setcounter`, `\addtocounter`
- #3: `\ltx@car` (for `\ref`), `\ltx@carsecond` (for `\pageref`)
- #4: LaTeX counter

```
\def\rc@@set#1#2#3#4{%
  \ifx#1\relax
    #2{#4}{\rc@default}%
  \else
    #2{#4}{%
      \expandafter#3#1\rc@default\rc@default\@nil
    }%
  \fi
}
```

2.5.2 User commands

\setcounterref

```
\rc@RobustDefZero\setcounterref{%
  \rc@set\setcounter\ltx@car
}
```

\addtocounterref

```
\rc@RobustDefZero\addtocounterref{%
  \rc@set\addtocounter\ltx@car
}
```

\setcounterpageref

```
\rc@RobustDefZero\setcounterpageref{%
  \rc@set\setcounter\ltx@carsecond
}
```

\addtocounterpageref

```
\rc@RobustDefZero\addtocounterpageref{%
  \rc@set\addtocounter\ltx@carsecond
}
```

2.6 Extracting references

\getrefnumber

```
\rc@newcommand*{\getrefnumber}[1]{%
  \romannumeral
  \ltx@ifundefined{r@#1}{%
    \expandafter\ltx@zero
    \rc@default
  }{%
    \expandafter\expandafter\expandafter\rc@extract@
    \expandafter\expandafter\expandafter!%
    \csname r@#1\expandafter\endcsname
    \expandafter{\rc@default}\@nil
  }%
}
```

\getpagerefnumber

```
\rc@newcommand*{\getpagerefnumber}[1]{%
  \romannumeral
  \ltx@ifundefined{r@#1}{%
    \expandafter\ltx@zero
    \rc@default
  }{%
    \expandafter\expandafter\expandafter\rc@extract@page
    \expandafter\expandafter\expandafter!%
    \csname r@#1\expandafter\expandafter\expandafter\endcsname
    \expandafter\expandafter\expandafter{%
      \expandafter\rc@default
    \expandafter}\expandafter{\rc@default}\@nil
  }%
}
```

\getrefbykeydefault

```
\rc@newcommand*{\getrefbykeydefault}[2]{%
  \romannumeral
```

```
  \expandafter\rc@getrefbykeydefault
    \csname r@#1\expandafter\endcsname
    \csname rc@extract@#2\endcsname
}
```

`\rc@getrefbykeydefault`

#1: `\r@<...>`
#2: `\rc@extract@<...>`
#3: default

```
\long\def\rc@getrefbykeydefault#1#2#3{%
  \ifx#1\relax
    % reference is undefined
    \ltx@ReturnAfterElseFi{%
      \ltx@zero
      #3%
    }%
  \else
    \ltx@ReturnAfterFi{%
      \ifx#2\relax
        % extract method is missing
        \ltx@ReturnAfterElseFi{%
          \ltx@zero
          #3%
        }%
      \else
        \ltx@ReturnAfterFi{%
          \expandafter
          \rc@generic#1{#3}{#3}{#3}{#3}{#3}{#3}\@nil#2{#3}%
        }%
      \fi
    }%
  \fi
}
```

`\rc@generic`

#1: first item in `\r@<...>`
#2: remaining items in `\r@<...>`
#3: `\rc@extract@<...>`
#4: default

```
\long\def\rc@generic#1#2\@nil#3#4{%
  #3{#1\TR@TitleReference\@empty{#4}\@nil}{#1}#2\@nil
}
```

`\rc@extract@`

```
\long\def\rc@extract@#1#2#3\@nil{%
  \ltx@zero
  #2%
}
```

`\rc@extract@page`

```
\long\def\rc@extract@page#1#2#3#4\@nil{%
  \ltx@zero
  #3%
}
```

`\rc@extract@name`

```
\long\def\rc@extract@name#1#2#3#4#5\@nil{%
  \ltx@zero
  #4%
}
```

\rc@extract@anchor

```
\long\def\rc@extract@anchor#1#2#3#4#5#6\@nil{%
  \ltx@zero
  #5%
}
```

\rc@extract@url

```
\long\def\rc@extract@url#1#2#3#4#5#6#7\@nil{%
  \ltx@zero
  #6%
}
```

\rc@extract@title

```
\long\def\rc@extract@title#1#2\@nil{%
  \rc@@extract@title#1%
}
```

\rc@@extract@title

```
\long\def\rc@@extract@title#1\TR@TitleReference#2#3#4\@nil{%
  \ltx@zero
  #3%
}
```

2.7 Macros for checking undefined references

\IfRefUndefinedExpandable

```
\rc@newcommand*{\IfRefUndefinedExpandable}[1]{%
  \ltx@ifundefined{r@#1}\ltx@firstoftwo\ltx@secondoftwo
}
```

\IfRefUndefinedBabel

```
\rc@RobustDefOne{}*\IfRefUndefinedBabel{%
  \begingroup
    \csname safe@actives@true\endcsname
  \expandafter\expandafter\expandafter\endgroup
  \expandafter\ifx\csname r@#1\endcsname\relax
    \expandafter\ltx@firstoftwo
  \else
    \expandafter\ltx@secondoftwo
  \fi
}
```

```
\rc@AtEnd%
⟨/package⟩
```

3 Installation

3.1 Download

Package. This package is available on CTAN[1]:

CTAN:macros/latex/contrib/refcount/refcount.dtx The source file.

CTAN:macros/latex/contrib/refcount/refcount.pdf Documentation.

[1] CTAN:pkg/refcount

Bundle. All the packages of the bundle 'oberdiek' are also available in a TDS compliant ZIP archive. There the packages are already unpacked and the documentation files are generated. The files and directories obey the TDS standard.

CTAN:install/macros/latex/contrib/refcount.tds.zip

TDS refers to the standard "A Directory Structure for TEX Files" (CTAN:pkg/tds). Directories with `texmf` in their name are usually organized this way.

3.2 Bundle installation

Unpacking. Unpack the `oberdiek.tds.zip` in the TDS tree (also known as `texmf` tree) of your choice. Example (linux):

```
unzip oberdiek.tds.zip -d ~/texmf
```

3.3 Package installation

Unpacking. The `.dtx` file is a self-extracting docstrip archive. The files are extracted by running the `.dtx` through plain TEX:

```
tex refcount.dtx
```

TDS. Now the different files must be moved into the different directories in your installation TDS tree (also known as `texmf` tree):

```
refcount.sty → tex/latex/refcount/refcount.sty
refcount.pdf → doc/latex/refcount/refcount.pdf
refcount.dtx → source/latex/refcount/refcount.dtx
```

If you have a `docstrip.cfg` that configures and enables docstrip's TDS installing feature, then some files can already be in the right place, see the documentation of docstrip.

3.4 Refresh file name databases

If your TEX distribution (TEX Live, MiKTEX, ...) relies on file name databases, you must refresh these. For example, TEX Live users run `texhash` or `mktexlsr`.

3.5 Some details for the interested

Unpacking with LATEX. The `.dtx` chooses its action depending on the format:

plain TEX: Run docstrip and extract the files.

LATEX: Generate the documentation.

If you insist on using LATEX for docstrip (really, docstrip does not need LATEX), then inform the autodetect routine about your intention:

```
latex \let\install=y\input{refcount.dtx}
```

Do not forget to quote the argument according to the demands of your shell.

Generating the documentation. You can use both the `.dtx` or the `.drv` to generate the documentation. The process can be configured by the configuration file `ltxdoc.cfg`. For instance, put this line into this file, if you want to have A4 as paper format:

```
\PassOptionsToClass{a4paper}{article}
```

An example follows how to generate the documentation with pdfLaTeX:

```
pdflatex refcount.dtx
makeindex -s gind.ist refcount.idx
pdflatex refcount.dtx
makeindex -s gind.ist refcount.idx
pdflatex refcount.dtx
```

4 History

[1998/04/08 v1.0]

- First public release, written as answer in the newsgroup comp.text.tex: "Re: Adding a \ref to a counter?"[2]

[2000/09/07 v2.0]

- Documentation added.
- LPPL 1.2
- Package rewritten, new commands added.

[2006/02/20 v3.0]

- Support for hyperref and nameref improved.
- Support for titleref and babel's shorthands added.
- New: `\refused`, `\getrefbykeydefault`

[2008/08/11 v3.1]

- Code is not changed.
- URLs updated.

[2010/12/01 v3.2]

- `\IfRefUndefinedExpandable` and `\IfRefUndefinedBabel` added.
- `\getrefnumber`, `\getpagerefnumber`, `\getrefbykeydefault` are expandable in exact two expansion steps.
- Non-expandable macros are made robust.
- Test files added.

[2] Url: https://groups.google.com/group/comp.text.tex/msg/c3f2a135ef5ee528

[2011/06/22 v3.3]

- Bug fix: `\rc@refused` is undefined for `\setcounterpageref` and similar macros. (Bug found by Marc van Dongen.)

[2011/10/16 v3.4]

- Bug fix: `\setcounterpageref` and `\addtocounterpageref` fixed. (Bug found by Staz.)
- Macros `\(setaddto)counter(page)ref` are made robust.

[2016/05/16 v3.5]

- Documentation updates.

[2019/12/15 v3.6]

- Documentation updates.

5 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; plain numbers refer to the code lines where the entry is used.