The uniquecounter package

Heiko Oberdiek*

2019/12/15 v1.4

Abstract

This package provides a kind of counter that provides unique number values. Several counters can be created by different names. The numeric values are not limited.

Contents

1 Documentation

`\UniqueCounterNew` {⟨*name*⟩}

Macro `\UniqueCounterNew` creates a new unique counter ⟨*name*⟩. An error is thrown, if the counter already exists.

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

`\UniqueCounterCall` {⟨*name*⟩} {⟨*code*⟩}

Macro `\UniqueCounterCall` calls the given ⟨*code*⟩ with a new value of counter ⟨*name*⟩ as argument.

`\UniqueCounterIncrement` {⟨*name*⟩}

Macro `\UniqueCounterIncrement` generates a new value for the counter ⟨*name*⟩ by incrementing by one (globally).

`\UniqueCounterGet` {⟨*name*⟩}

Expandable macro `\UniqueCounterGet` returns the current value of counter ⟨*name*⟩

1.1 Example

```
⟨*example⟩
\documentclass{minimal}
\usepackage{uniquecounter}
\UniqueCounterNew{anchor}
\makeatletter
\newcommand*{\DefNewAnchorName}[2]{%
  % #1 is unique counter value
  % #2 is name of anchor
  \@namedef{anchor@#2}{a#1}%
}
\newcommand*{\NewAnchorName}[1]{%
  \UniqueCounterCall{anchor}\DefNewAnchorName{#1}%
}
\newcommand*{\PrintAnchorName}[1]{%
  \@nameuse{anchor@#1}%
}
\begin{document}
  \NewAnchorName{Top}%
  \NewAnchorName{Left}%
  \noindent
  Top: \PrintAnchorName{Top}\\%
  Left: \PrintAnchorName{Left}%
\end{document}
⟨/example⟩
```

2 Implementation

```
⟨*package⟩
```

2.1 Reload check and package identification

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

```
\begingroup\catcode61\catcode48\catcode32=10\relax%
  \catcode13=5 % ^^M
  \endlinechar=13 %
  \catcode35=6 % #
  \catcode39=12 % '
  \catcode44=12 % ,
  \catcode45=12 % -
  \catcode46=12 % .
```

```
\catcode58=12 % :
\catcode64=11 % @
\catcode123=1 % {
\catcode125=2 % }
\expandafter\let\expandafter\x\csname ver@uniquecounter.sty\endcsname
\ifx\x\relax % plain-TeX, first loading
\else
  \def\empty{}%
  \ifx\x\empty % LaTeX, first loading,
    % variable is initialized, but \ProvidesPackage not yet seen
  \else
    \expandafter\ifx\csname PackageInfo\endcsname\relax
      \def\x#1#2{%
        \immediate\write-1{Package #1 Info: #2.}%
      }%
    \else
      \def\x#1#2{\PackageInfo{#1}{#2, stopped}}%
    \fi
    \x{uniquecounter}{The package is already loaded}%
    \aftergroup\endinput
  \fi
\fi
\endgroup%
```

Package identification:

```
\begingroup\catcode61\catcode48\catcode32=10\relax%
  \catcode13=5 % ^^M
  \endlinechar=13 %
  \catcode35=6 % #
  \catcode39=12 % '
  \catcode40=12 % (
  \catcode41=12 % )
  \catcode44=12 % ,
  \catcode45=12 % -
  \catcode46=12 % .
  \catcode47=12 % /
  \catcode58=12 % :
  \catcode64=11 % @
  \catcode91=12 % [
  \catcode93=12 % ]
  \catcode123=1 % {
  \catcode125=2 % }
  \expandafter\ifx\csname ProvidesPackage\endcsname\relax
    \def\x#1#2#3[#4]{\endgroup
      \immediate\write-1{Package: #3 #4}%
      \xdef#1{#4}%
    }%
  \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@uniquecounter.sty\endcsname
```

```
\ProvidesPackage{uniquecounter}%
  [2019/12/15 v1.4 Provide unlimited unique counter (HO)]%
```

2.2 Catcodes

```
\begingroup\catcode61\catcode48\catcode32=10\relax%
  \catcode13=5 % ^^M
  \endlinechar=13 %
  \catcode123=1 % {
  \catcode125=2 % }
  \catcode64=11 % @
  \def\x{\endgroup
    \expandafter\edef\csname uqc@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\uqc@AtEnd{%
    \uqc@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{43}{12}% +
\TMP@EnsureCode{46}{12}% .
\TMP@EnsureCode{47}{12}% /
\TMP@EnsureCode{91}{12}% [
\TMP@EnsureCode{93}{12}% ]
\TMP@EnsureCode{96}{12}% `
\edef\uqc@AtEnd{\uqc@AtEnd\noexpand\endinput}
```

```
\begingroup\expandafter\expandafter\expandafter\endgroup
\expandafter\ifx\csname RequirePackage\endcsname\relax
  \def\TMP@RequirePackage#1[#2]{%
    \begingroup\expandafter\expandafter\expandafter\endgroup
    \expandafter\ifx\csname ver@#1.sty\endcsname\relax
      \input #1.sty\relax
    \fi
  }%
  \TMP@RequirePackage{bigintcalc}[2007/11/11]%
  \TMP@RequirePackage{infwarerr}[2007/09/09]%
\else
```

```
  \RequirePackage{bigintcalc}[2007/11/11]%
  \RequirePackage{infwarerr}[2007/09/09]%
\fi
```

`\uqc@IncNum`

```
\begingroup\expandafter\expandafter\expandafter\endgroup
\expandafter\ifx\csname numexpr\endcsname\relax
  \def\uqc@IncNum#1{%
    \begingroup
      \count@=\csname uqc@cnt@#1\endcsname\relax
      \advance\count@\@ne
      \expandafter\xdef\csname uqc@cnt@#1\endcsname{%
        \number\count@
      }%
      \ifnum\count@=2147483647 %
        \global\expandafter\let\csname uqc@inc@#1\endcsname
        \uqc@IncBig
      \fi
    \endgroup
  }%
\else
  \def\uqc@IncNum#1{%
    \expandafter\xdef\csname uqc@cnt@#1\endcsname{%
      \number\numexpr\csname uqc@cnt@#1\endcsname+1%
    }%
    \ifnum\csname uqc@cnt@#1\endcsname=2147483647 %
      \global\expandafter\let\csname uqc@inc@#1\endcsname
      \uqc@IncBig
    \fi
  }%
\fi
```

`\uqc@IncBig`

```
\def\uqc@IncBig#1{%
  \expandafter\xdef\csname uqc@cnt@#1\endcsname{%
    \expandafter\expandafter\expandafter
    \BigIntCalcInc\csname uqc@cnt@#1\endcsname!%
  }%
}
```

`\uqc@Def`

```
\begingroup\expandafter\expandafter\expandafter\endgroup
\expandafter\ifx\csname newcommand\endcsname\relax
  \def\uqc@Def#1{\def#1##1}%
\else
  \def\uqc@Def#1{\newcommand*{#1}[1]}%
\fi
```

`\UniqueCounterNew`

```
\uqc@Def\UniqueCounterNew{%
  \expandafter\ifx\csname uqc@cnt@#1\endcsname\relax
    \expandafter\xdef\csname uqc@cnt@#1\endcsname{0}%
    \global\expandafter\let\csname uqc@inc@#1\endcsname\uqc@IncNum
    \@PackageInfo{uniquecounter}{New unique counter `#1'}%
  \else
    \@PackageError{uniquecounter}{Unique counter `#1' is already defined}\@ehc
  \fi
}
```

\UniqueCounterIncrement

```
\uqc@Def\UniqueCounterIncrement{%
  \expandafter\ifx\csname uqc@cnt@#1\endcsname\relax
    \@PackageError{uniquecounter}{Unique counter `#1' is undefined}\@ehc
  \else
    \csname uqc@inc@#1\endcsname{#1}%
  \fi
}
```

\UniqueCounterGet

```
\uqc@Def\UniqueCounterGet{%
  \csname uqc@cnt@#1\endcsname
}
```

\UniqueCounterCall

```
\uqc@Def\UniqueCounterCall{%
  \expandafter\ifx\csname uqc@cnt@#1\endcsname\relax
    \@PackageError{uniquecounter}{Unique counter `#1' is undefined}\@ehc
    \expandafter\uqc@Call\expandafter0%
  \else
    \UniqueCounterIncrement{#1}%
    \expandafter\expandafter\expandafter\uqc@Call
    \expandafter\expandafter\expandafter{%
      \csname uqc@cnt@#1\expandafter\endcsname\expandafter
    }%
  \fi
}
```

\uqc@Call

```
\long\def\uqc@Call#1#2{#2{#1}}%
```

```
\uqc@AtEnd%
```

⟨/package⟩

3 Installation

3.1 Download

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

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

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

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/uniquecounter.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.

[1] CTAN:pkg/uniquecounter

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 uniquecounter.dtx
```

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

```
uniquecounter.sty         → tex/generic/uniquecounter/uniquecounter.sty
uniquecounter.pdf         → doc/latex/uniquecounter/uniquecounter.pdf
uniquecounter-example.tex → doc/latex/uniquecounter/uniquecounter-example.tex
uniquecounter.dtx         → source/latex/uniquecounter/uniquecounter.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{uniquecounter.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 uniquecounter.dtx
makeindex -s gind.ist uniquecounter.idx
pdflatex uniquecounter.dtx
makeindex -s gind.ist uniquecounter.idx
pdflatex uniquecounter.dtx
```

4 History

[2009/09/11 v1.0]

- First public version.

[2009/12/18 v1.1]

- Bug fix in `\UniqueCounterCall` for values > 9 (bug report of Lev Bishop).

[2011/01/30 v1.2]

- Already loaded package files are not input in plain TEX.

[2016/05/16 v1.3]

- Documentation updates.

[2019/12/15 v1.4]

- 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.

P

R

T

U

W

X