

cntperchap.sty

**Store counter values per chapter
(or other section levels)**

Version 0.2

05/09/2015

Author: Christian Hupfer[†]

Contents

1	Disclaimer	3
2	Introduction	3
2.1	Basic usage	4
3	Package options	4
4	Requirements and incompatibilities	5
4.1	Required packages	5
4.2	Incompatibilities	5
5	Documentation of Macros	5
5.1	Preamble only commands	5
5.2	User macros	7
6	Informational macros	8
7	Conditionals	9
8	Generated files	10
9	To-Do list	10

[†]christian.hupfer@yahoo.de

10 Acknowledgments	11
11 Version history	12
Index	13



1 Disclaimer

This package as of its version 0.2 is under constant development and as such subject to macro interface changes as well as renaming of macros.

2 Introduction

The aim of this package is to provide support for a summary in advance how many sections, subsections, etc. or figures, tables, equations there will be in predefined section level, for example per chapters. The values are stored at the beginning of a new chapter and written to `\jobname.cpsfoo` files, where `foo` is short for the relevant counter name. The section level can be one of standard levels

- `\part`
- `\chapter`
- `\section`
- `\subsection`
- `\subsubsection`
- `\paragraph`
- `\subparagraph`

The section level can be specified as an option (see 3), but is preconfigured on the underlying document class. For  `book`-like classes, the section level defaults to `\chapter`, for  `article`-like documentclasses it defaults to `\section`.

This package is the consequence of the question <http://tex.stackexchange.com/questions/241559/how-to-count-the-total-number-of-sections-within-a-chapter> by the user `gsl`.

Statistics of the local chapter

```
section 1 has 0 page(s)
section 1 has 1 section(s)
section 1 has 0 subsection(s)
section 1 has 0 subsubsection(s)
section 1 has 0 table(s)
section 1 has 0 figure(s)
```

2.1 Basic usage



If `\nofiles` is used, no package related files are generated or included, similar to the behaviour for `\tableofcontents` etc.

```
\documentclass{book}
\usepackage{cntperchap}

\RegisterCounters{page,section,subsection,subsubsection,table,figure}

\begin{document}
\section{First section}
\ShowStatistics
\end{document}
```

3 Package options

As of version 0.2 the package has only following options so far:

- **autodefine**
This option enables the automatic counter definition if some undefined counter is specified during the registration process.
- **draft** (default is draft)
This option enables the explicit statistics
- **final** (default is off)
This option disables the explicit statistics
- **noendclose** (default is false)
This option prevents the explicit closing of output files and provides the means to enter a hook into the `\AfterEndDocument` command.
- **tracklevel** $\langle =chapter \rangle$ (chapter)
This option sets the sectioning level under supervision (tracked). If not specified, the counters are stored on a per chapter base, if set to **section**, the counter values are stored on a per section level etc. The section level has to be specified without the backslash `\`.
Unknown section levels will lead to a compilation error.

- **verbose**

This option enables verbose output during compilation.

4 Requirements and incompatibilities

4.1 Required packages

- `\assocnt`
- `\etoolbox`
- `\ifthen`
- `\morewrites`
- `\xkeyval`
- `\xparse`

The packages `\etoolbox` and `\xkeyval` are already loaded by `\assocnt`.

4.2 Incompatibilities

This package has been tested with the standard classes `\article`, `\book` and `\report` as well as with `\memoir` and the relevant `\KOMA` equivalents. As of version 0.2 for those classes there are no known incompatibilities with the general behaviour of the package, however, there is an issue with `\assocnt` and `\xifthen` which is not solved so far.

It would be nice to adapt the package for usage on a per frame base with the `\beamer` class, see 9

5 Documentation of Macros

5.1 Preamble only commands

`\RegisterCounter[<options>]{<counter>}`

This provides the means to let the package know that a counter should be tracked for values per chapter

Description of arguments of command `\RegisterCounter`

#1 [*options*]:

`autodefine` $\langle true, false \rangle$

This will enable the automatic definition of a counter if the name specified as 2nd mandatory argument is not identified as L^AT_EX counter.

#2 $\{\langle counter \rangle\}$ This contains the counter name which should be tracked for counting itself.

`\RegisterCounters` $\{\langle counter1, counter2, \dots \rangle\}$

This macro allows for specification of a CSV list of counter names – internally this will call `\RegisterCounter`^{→ P. 5}

Description of arguments of command `\RegisterCounters`

- $\{\langle counter1, counter2, \dots \rangle\}$

This contains a comma separated list of counter names which should be tracked for counting itself.

Please note, that both `\RegisterCounter`^{→ P. 5} and `\RegisterCounters` can be used only in the preamble of a document.

It is safe to use `\RegisterCounters` with just one counter name.

`\PrepareTocCommand` $\{\langle toc command \rangle\}$

This command disables the usage of the automatic counter storage for ToC - like macros (`\tableofcontents`, `\listoffigures`, `\listoftables`, since those use `\chapter*` and confuse the tracking counter thereby.

Description of arguments of command `\PrepareTocCommand`

$\{\langle toc command \rangle\}$ This is intended to be a command such as `\tableofcontents` etc., but it could be any command actually.

Please use this command with care!

By default, `\tableofcontents`, `\listoffigures`, `\listoftables` and `\printindex` are used with this feature.

5.2 User macros

`\GetTotalCounter` [*`<chapter number>`*] {*`<counter>`*}

This returns the total value of the counter (specified as 2nd argument) per chapter. This macro should be expandable.

Description of arguments of command `\GetTotalCounter`

- [*`<chapter number>`*] This optional argument contains the number of the chapter from which the counter value should be extracted. If this is not given, the current chapter is used.
- {*`<counter>`*} This is the name of of the counter whose per chapter value should be displayed.

`\IndividualCounterStatistics` [*`<chapter number>`*] {*`<counter>`*}

This returns the total value of the counter (specified as 2nd argument) per chapter.

Description of arguments of command `\IndividualCounterStatistics`

- [*`<chapter number>`*] This optional argument contains the number of the chapter from which the counter value should be extracted. If this is not given, the current chapter is used.
- {*`<counter>`*} This is the name of of the counter whose per chapter value should be displayed together with the counter name.

`\ShowStatistics` [*`<chapter number>`*]

This returns the all total values of the counters registered by `\RegisterCounter`^{→P.5} or `\RegisterCounters`^{→P.6} per chapter.

Description of arguments of command `\ShowStatistics`

- [*`<chapter number>`*] This optional argument contains the number of the chapter from which the counter value should be extracted. If this is not given, the current chapter is used.

Both `\ShowStatistics` and `\IndividualCounterStatistics` are defined to provide nothing if `final`^{→P.4} is given as package option.

`\Fullstatistics`

New: 2015-05-07

This will show an overview of all registered counters throughout the document (unless storage was locally disabled).

`\Fullstatistics`

This document has
 13 page(s)
 11 section(s)
 5 subsection(s)
 0 subsubsection(s)
 0 table(s)
 0 figure(s)

`\StoreCounters`

This macro stores all registrated counter values to the file.

Remark: This command was named `\StoreCountersPerChapter` in previous versions

`\StopCounting`

This macro stores all registrated counter values to the file.

This command is automatically called at the end of the document, but can be used before too. It is useful, if there is some `\printindex` etc. at the end and only one chapter. In this case, the finalizing storage would not be done automatically.

`\RestoreCounter`{*<counter>*}

This macro restores the counter `{<counter>}` to zero.

It is not necessary to use this macro directly in almost any case.

`\RestoreCounters`

This macro restores all registrated counters to zero. This happens every time if a new driving section level is used.

6 Informational macros

`\numberofstoredcounters`

This macro just returns the number of registered counters. It will not return the internal counter name which holds the number.

7 Conditionals

The package knows some conditionals:

`\ifcpsstorage`

New: 2015-05-08

This status variable enables or disables the storage of counters per chapter (section,...). It is a classical T_EX `\newif` statement and can be set to true by saying `\cpsstoragetrue` or false `\cpsstorefalse`. However, there are two small macros to wrap this enabling or disabling into to more memorable names: `\EnableCPSStorage` and `\DisableCPSStorage`.

`\EnableCPSStorage`

New: 2015-05-08

Enables the storage of counters per chapter (etc.)

`\DisableCPSStorage`

New: 2015-05-08

Enables the storage of counters per chapter (etc.)

`\ifcpsdraftmode`


New: 2015-05-03

This is a classical T_EX conditional defined by `\newif`. It can be used to enable draft mode locally by saying `\cpsdraftmodetrue` and disable in return using `\cpsdraftmodefalse`.

8 Generated files

The package uses the same approach as \LaTeX commands to write content to the table of content files. It creates `.cpsfoo` files, where `\foo` is the name of the relevant counter. Those files are constantly updated if the main document has been changed. It's safe to delete those files, however, the correct values appear only after the 2nd and consecutive runs.

9 To-Do list

- Better error handling (no checks for many features so far).
- Squeeze the usage of multiple files for each counter value into a more sophisticated approach.
- More options for fine control of the behaviour package and macros.
- Easy - adaption for other documentclasses, especially for  **beamer**
- Improve documentation

If you

- find bugs
- errors in the documentation
- have suggestions
- have feature requests


don't hesitate and contact me via christian.hupfer@yahoo.de

10 Acknowledgments

I would like to express my gratitudes to the developpers of fine L^AT_EX packages and of course to the users at `tex.stackexchange.com`, especially to

- Enrico Gregorio
- Joseph Wright
- David Carlisle
- Paulo Roberto Massa Cereda
- Werner Doe (;-))
- Johannes Böttcher (for some remarks/checking on KOMA related issues)

for their invaluable help on many questions on macros.

A special gratitude goes to Prof. Dr. Dr. Thomas Sturm for providing the wonderful  `tcolorbox` package which was used to write this documentation.

11 Version history

- 0.2:
 - Introduced the possibility to disable the storage temporarily
 - Fixed a severe bug concerning the file handle generation
 - Added support for ‘tracklevel=option’, allowing for arbitrary section levels as watch
 - Renamed some macros which had ‘Chapter’ in name
 - Added a check if the relevant section level sequence exists at all
 - Added a total counter for the section level and limited the ‘\GetTotalCounters’ max value
 - Fixed typos in documentation
- 0.1: Initial version, fixed errors with `\dolistcsloop` for getting the counter values and replaced it with `\forlistcsloop`
- 0.01: Bootstrap version

Index

autodefine key, 4, 6

Class

- article, 3, 5
- beamer, 5, 10
- book, 3, 5
- KOMA, 5
- memoir, 5
- report, 5

Counters

- foo, 3

\DisableCPSStorage, 9

draft key, 4

\EnableCPSStorage, 9

final key, 4

foo counter, 3

\Fullstatistics, 7

\GetTotalCounter, 7

\ifcpsdraftmode, 9

\ifcpsstorage, 9

\IndividualCounterStatistics, 7

Keys

- autodefine, 4, 6
- draft, 4
- final, 4
- noendclose, 4
- tracklevel, 4
- verbose, 5

noendclose key, 4

\numberofstoredcounters, 8

Package

- assocnt, 5
- etoolbox, 5
- ifthen, 5
- morewrites, 5

tcolorbox, 11

xifthen, 5

xkeyval, 5

xparse, 5

\PrepareTocCommand, 6

\RegisterCounter, 5

\RegisterCounters, 6

\RestoreCounter, 8

\RestoreCounters, 8

\ShowStatistics, 7

\StopCounting, 8

\StoreCounters, 8

tracklevel key, 4

verbose key, 5