

The textualcomma package

<https://github.com/SFr682k/textualcomma>

Sebastian Friedl
sfr682k@t-online.de

2017/12/03

Abstract

A solution for using the textual comma character as decimal separator in math mode.

textualcomma is based on the icomma package and intended as a solution for situations where the text comma character discerns from the math comma character. Escaping to text mode every time a comma is used in math mode may slow down the compilation process.

Please check, whether icomma already satisfies your needs.

Obtaining the package and documentation file

You can obtain the package files by running \LaTeX over the file textualcomma.ins:
`latex textualcomma.ins`

For compiling the documentation, run **Lua \LaTeX** (!) twice over textualcomma.dtx:

```
lualatex textualcomma.dtx
```

```
lualatex textualcomma.dtx
```

Please verify that all dependencies are fulfilled *before* compiling.

License

© 2017 Sebastian Friedl

This work may be distributed and/or modified under the conditions of the \LaTeX Project Public License, either version 1.3c of this license or (at your option) any later version.

The latest version of this license is available at <http://www.latex-project.org/lppl.txt> and version 1.3c or later is part of all distributions of \LaTeX version 2008-05-04 or later.

This work has the LPPL maintenance status ‘maintained’. The current maintainer of this work is Sebastian Friedl.

This work consists of the files textualcomma.dtx and textualcomma.ins and the derived file textualcomma.sty.

Call for cooperation

Please report bugs and other problems as well as suggestions for improvements by using the issue tracker on GitHub or sending an email to sfr682k@t-online.de.

Part I

The documentation

1 Introduction

Using the comma as decimal separator in ordinary \LaTeX has always been problematic, since the comma is treated as a punctuation character and \LaTeX adds some space after it.

The solution was defining an ‘intelligent’ comma ‘recognizing’ whether it is used as decimal separator or punctuation character.

The common solution to this problem is the `icomma` package by Walter Schmidt [1]. However, some fonts, e. g. Source Serif Pro, don’t provide any math support. Although there are fonts looking very similar to Source Serif and providing math support (like Utopia and its ‘clones’ Fourier and Erewhon), more or less obvious differences are visible (compare ‘,’ to ‘,’). The only solution solving this ‘problem’ is escaping to text mode.

2 The `textualicomma` package

2.1 `textualicomma`’s behaviour

`textualicomma` makes some changes and additions to `icomma`’s code in order to provide automated escaping to text mode. The behaviour remains the same:

- The comma is treated as decimal separator if it is directly followed by a number. Thus, a number is to be entered as, for instance,
`3,142`
- whereas mathematical expressions like (x, y) are to be written with a space after the comma:
`(x, \, y)`

Also, the period is redefined so that it adapts to the used font.

2.2 Package options

By default, the font used for typesetting the comma is the roman text font. This behaviour can be modified by using package options:

`sffamily` Passing the `sffamily` option to `textualicomma` results in the comma being printed using the loaded sans serif font

`ttfamily` Passing the `ttfamily` option to `textualicomma` results in the comma being printed using the loaded typewriter font

`\textualicommafont` Both options change the used font *for the whole document*. Changes inside a document can be made using the `\textualicommafont` command:

```
\textualicommafont{\rmfamily}  
\textualicommafont{\sffamily}  
\textualicommafont{\ttfamily}
```

3 Bugs and problems

Generally, since the comma is made ‘active’, problems are not unlikely.

For example, when using the ‘intelligent comma’ together with the `dcolumn` package, a comma to be *printed* as the decimal separator in a column of type D is to be specified as `{\mathord\mathcomma}`, rather than `{,}`, since the latter leads to an error. For instance:

```
\begin{tabular}{... D{,}{\mathord\mathcomma}{2} ...}
```

Note that specifying the comma as the related input character works as usual.

References

- [1] Walter Schmidt. The `icomma` package for \LaTeX 2 ϵ . Package and Documentation available on CTAN (<https://www.ctan.org/pkg/icomma>)

Part II

The package code

```
1 \ProvidesPackage{textualicomma}[2017/12/03 textualicomma package]
```

Use the roman text font family by default

```
2 \def\mathtxtcmmafffamily{\rmfamily}
```

Provide an option for switching to the sans serif font

```
3 \newif\if@mathtxtcmmasfffamily
4 \@mathtxtcmmasfffamilyfalse
5 \DeclareOption{sffamily}{\@mathtxtcmmasfffamilytrue}
```

Provide an option for switching to the typewriter font

```
6 \newif\if@mathtxtcmmattfamily
7 \@mathtxtcmmattfamilyfalse
8 \DeclareOption{ttfamily}{\@mathtxtcmmattfamilytrue}
```

Process all defined options

```
9 \ProcessOptions
```

If one of the sffamily or ttfamily options has been used, changes are applied

```
10 \if@mathtxtcmmasfffamily\def\mathtxtcmmafffamily{\sffamily}\fi
11 \if@mathtxtcmmattfamily\def\mathtxtcmmafffamily{\ttfamily}\fi
```

Provide an command for switching the font inside the document

```
12 \def\textualicommafont#1{\def\mathtxtcmmafffamily{#1}}
```

We define the comma character at the beginning of the document.

```
13 \AtBeginDocument{%
14   \def\mathtextcomma{\text{\mathtxtcmmafffamily},}
```

I don't know what the following lines are exactly doing, but they can't be removed; probably, because they are making the comma 'active'

```
15   \mathchardef\mathcomma\mathcode`\,\,%
16   \mathcode`\,="8000 %
17 }
18 {\catcode`\,=\active
19  \gdef,{\futurelet@let@token\sm@rtcomma}
20 }
```

Define the 'smart comma'

```
21 \def\sm@rtcomma{%
22   \ifx\@let@token@sptoken\mathtextcomma\protect\,\else
23   \ifx\@let@token\space \else
24     \mathtextcomma\fi\fi}
```

Allow changing of the period's font.

```
25 \DeclareMathSymbol{.}{\mathalpha}{operators}{`.`}
```

That's enough code. \endinput.

```
26 \endinput
```