

The **telprint** package

Heiko Oberdiek
<oberdiek@uni-freiburg.de>

2006/08/16 v1.7

Abstract

Package **telprint** provides `\telprint` for formatting German phone numbers.

Contents

1	Documentation	1
1.1	Introduction	1
1.2	Short overview in English	2
1.2.1	Configuration	2
1.3	Documentation in German	2
2	Implementation	3
2.1	Reload check and package identification	3
3	Installation	6
3.1	Download	6
3.2	Bundle installation	6
3.3	Package installation	6
3.4	Refresh file name databases	7
3.5	Some details for the interested	7
4	History	7
	[1996/11/28 v1.0]	7
	[1997/09/16 v1.1]	7
	[1997/10/16 v1.2]	7
	[1997/12/09 v1.3]	8
	[2004/11/02 v1.4]	8
	[2005/09/30 v1.5]	8
	[2006/02/12 v1.6]	8
	[2006/08/26 v1.7]	8
5	Index	8

1 Documentation

1.1 Introduction

This is a very old package that I have written to format phone numbers. It follows German conventions and the documentation is mainly in German.

1.2 Short overview in English

L^AT_EX:

```
\usepackage{telprint}
\telprint{123/456-789}
```

plain-T_EX:

```
\input telprint.sty
\telprint{123/456-789}
```

\telprint **\telprint{...}** formats the explicitly given number. Digits, spaces and some special characters ('+', '/', '-', '(', ')', '~', ' ') are supported. Numbers are divided into groups of two digits from the right. Examples:

```
\telprint{0761/12345}      ==> 07\,61/1\,23\,45
\telprint{01234/567-89}    ==> 0\,12\,34/5\,67\leavevmode\hbox{-}89
\telprint{+49 (6221) 297}  ==> +49~(62\,21)~2\,97
```

1.2.1 Configuration

The output of the symbols can be configured by **\telhyphen**, **\telslash**, **\telleftparen**, **\telrightparen**, **\telplus**, **\teltilde**. Example:

```
\telslash{\,/,\,}\ \telprint{12/34} ==> 12\,/,\,34
```

\telspace **\telspace** configures the space between digit groups.

\telnumber **\telnumber** only formats a number in digit groups; special characters are not recognized.

1.3 Documentation in German

\telprint • **telprint#1**
Der eigentliche Anwenderbefehl zur formatierten Ausgabe von Telefonnummern. Diese dürfen dabei nur als Zahlen angegeben werden(, da sie tokenweise analysiert werden). Als Trenn- oder Sonderzeichen werden unterstützt: '+', '/', '-', '(', ')', '~', ' '. Einfache Leerzeichen werden erkannt und durch Tilden ersetzt, um Trennungen in der Telefonnummer zu verhindern. (Man beachte aus gleichem Grunde die **\hbox** bei '-'). Beispiele:

```
\telprint{0761/12345}      ==> 07\,61/1\,23\,45
\telprint{01234/567-89}    ==> 0\,12\,34/5\,67\leavevmode\hbox{-}89
\telprint{+49 (6221) 297}  ==> +49~(62\,21)~2\,97
```

Der Rest enthält eher Technisches:

\telspace • **\telspace#1**
Mit diesem Befehl wird der Abstand zwischen den Zifferngruppen angegeben (Default: \,). (Durch **\telspace{}** kann dieser zusätzliche Abstand abgestellt werden.)

\telhyphen • **\telhyphen#1**
Dieser Befehl gibt die Art des Bindestriches, wie er ausgegeben werden soll. In der Eingabe darf jedoch nur der einfache Bindestrich stehen: **\telprint{123-45}**, jedoch NIE **\telprint{123--45}**! Kopka-Bindestrich-Fans geben an: **\telhyphen{\leavevmode\hbox{--}}**

\telslash • **\telslash#1, \telleftparen#1, \telrightparen#1, \telplus#1, \teltilde**
\telleftparen Diese Befehle konfigurieren die Zeichen '/', '(', ')', '+' und '~'. Sie funktionieren analog zu **\telhyphen**.
\telrightparen
\telplus
\teltilde

- `\telnumber#1`
`\telnumber` Richtung interner Befehl: Er dient dazu, eine Zifferngruppe in Zweiergruppen auszugeben. Die einzelnen Zahlen werden im Tokenregister `\TELtoks` gespeichert. Abwechselnd werden dabei zwischen zwei Token (Zahlen) `\TELx` bzw. `\TELy` eingefuegt, abhängig von dem wechselnden Wert von `\TELswitch`. Zum Schluss kann dann einfach festgestellt werden ob die Nummer nun eine geradzahlige oder ungeradzahlige Zahl von Ziffern aufwies. Dem entsprechend wird `\TELx` mit dem Zusatzabstand belegt und `\TELy` leer definiert oder umgekehrt.)
- `\TEL...` interne Befehle, Technisches:
`\TELsplit` dient zur Aufteilung einer zusammengesetzten Telefonnummer (Vorwahl, Hauptnummer, Nebenstelle). In dieser Implementation werden als Trennzeichen nur `'/'` und `'-'` erkannt. Die einzelnen Bestandteile wie Vorwahl werden dann dem Befehl `\telnumber` zur Formatierung uebergeben.
- Die Erkennung von einfachen Leerzeichen ist um einiges schwieriger: Die Tokentrennung ueber Parameter `#1#2` funktioniert nicht für einfache Leerzeichen, da TeX sie *niemals* als eigenständige Argumente behandelt! (The TeXbook, Chapter 20, p. 201)
 (Anmerkung am Rande: Deshalb funktionieren die entsprechenden Tokenmakros auf S. 149 des Buches „Einführung in TeX“ von N. Schwarz (3. Aufl.) nicht, wenn im Tokenregister als erstes ein einfaches Leerzeichen steht!)

2 Implementation

1 `(*package)`

2.1 Reload check and package identification

Reload check, especially if the package is not used with L^AT_EX.

```

2 \begingroup
3 \expandafter\let\expandafter\x\csname ver@telprint.sty\endcsname
4 \ifcase 0%
5   \ifx\x\relax % plain
6   \else
7     \ifx\x\empty % LaTeX
8     \else
9       1%
10    \fi
11  \fi
12 \else
13   \expandafter\ifx\csname PackageInfo\endcsname\relax
14     \def\x#1#2{%
15       \immediate\write-1{Package #1 Info: #2.}%
16     }%
17   \else
18     \def\x#1#2{\PackageInfo{#1}{#2, stopped}}%
19   \fi
20   \x{telprint}{The package is already loaded}%
21 \endgroup
22 \expandafter\endinput
23 \fi
24 \endgroup

```

Package identification:

```

25 \begingroup
26 \expandafter\ifx\csname ProvidesPackage\endcsname\relax
27   \def\x#1#2#3[#4]{\endgroup
28     \immediate\write-1{Package: #3 #4}%
29     \xdef#1{#4}%

```

```

30     }%
31   \else
32     \def\x#1#2[#3]{\endgroup
33       #2[#3]}%
34     \ifx#1\relax
35       \xdef#1[#3]%
36     \fi
37   }%
38 \fi
39 \expandafter\x\csname ver@telprint.sty\endcsname
40 \ProvidesPackage{telprint}%
41   [2006/08/16 v1.7 Formatting of German phone numbers (HO)]
42 \ifx\telprint\UnDeFiNeD
43 % \fi in derselben Zeile wie \endinput oder \expandafter verwenden!
44 \else\endinput\fi
45 \ifx\DeclareRobustCommand\UnDeFiNeD
46   \def\DeclareRobustCommand*#1[1]{\def#1##1}
47   \def\TELreset{\let\DeclareRobustCommand=\UnDeFiNeD}
48   \immediate\write-1{Package 'telprint' warning: %
49     Macros are not robust!}
50 \else
51   \let\TELreset=\relax
52 \fi

\telspace

53 \DeclareRobustCommand*\telspace[1]{\def\TELspace{#1}}
54 \telspace{ }$,$}

\telhyphen

55 \DeclareRobustCommand*\telhyphen[1]{\def\TELhyphen{#1}}
56 \telhyphen{\leavevmode\hbox{-}}% \hbox zur Verhinderung der Trennung

\telslash

57 \DeclareRobustCommand*\telslash[1]{\def\TELslash{#1}}
58 \telslash{/}%

\telleftparen

59 \DeclareRobustCommand*\telleftparen[1]{\def\TELleftparen{#1}}
60 \telleftparen{ }%

\telrightparen

61 \DeclareRobustCommand*\telrightparen[1]{\def\TELrightparen{#1}}
62 \telrightparen{ }%

\telplus

63 \DeclareRobustCommand*\telplus[1]{\def\TELplus{#1}}
64 \telplus{+}%

\teltilde

65 \DeclareRobustCommand*\teltilde[1]{\def\TELtilde{#1}}
66 \teltilde{~}%

\TELtoks

67 \newtoks\TELtoks

\TELnumber

68 \def\TELnumber#1#2\TELnumberEND{%
69   \begingroup
70   \def\0{#2}%

```

```

71 \expandafter\endgroup
72 \ifx\0\empty
73   \TELtoks=\expandafter{\the\TELtoks#1}%
74   \ifnum\TELswitch=0
75     \def\TELx{\TELspace}\def\TELy{}%
76   \else
77     \def\TELx{}\def\TELy{\TELspace}%
78   \fi
79   \the\TELtoks
80 \else
81   \ifnum\TELswitch=0
82     \TELtoks=\expandafter{\the\TELtoks#1\TELx}%
83     \def\TELswitch{1}%
84   \else
85     \TELtoks=\expandafter{\the\TELtoks#1\TELy}%
86     \def\TELswitch{0}%
87   \fi
88   \TELnumber#2\TELnumberEND
89 \fi
90 }

\telnumber
91 \DeclareRobustCommand*\telnumber*[1]{%
92   \TELtoks={}%
93   \def\TELswitch{0}%
94   \TELnumber#1\TELnumberEND
95 }

\TELSplit
96 \def\TELSplit{\futurelet\TELfuture\TELDosplit}

\TELDosplit
97 \def\TELDosplit#1#2\TELSplitEND
98 {%
99   \def\TELsp{ }%
100  \expandafter\ifx\TELsp\TELfuture
101    \let\TELfuture=\relax
102    \expandafter\telnumber\expandafter{\the\TELtoks}~%
103    \telprint{#1#2}% Das Leerzeichen kann nicht #1 sein!
104  \else
105    \def\TELfirst{#1}%
106    \ifx\TELfirst\empty
107      \expandafter\telnumber\expandafter{\the\TELtoks}%
108      \TELtoks={}%
109    \else\if-\TELfirst
110      \expandafter\telnumber\expandafter{\the\TELtoks}\TELhyphen
111      \telprint{#2}%
112    \else\if/\TELfirst
113      \expandafter\telnumber\expandafter{\the\TELtoks}\TELslash
114      \telprint{#2}%
115    \else\if(\TELfirst
116      \expandafter\telnumber\expandafter{\the\TELtoks}\TELleftparen
117      \telprint{#2}%
118    \else\if)\TELfirst
119      \expandafter\telnumber\expandafter{\the\TELtoks}\TELrightparen
120      \telprint{#2}%
121    \else\if+\TELfirst
122      \expandafter\telnumber\expandafter{\the\TELtoks}\TELplus
123      \telprint{#2}%
124    \else\def\TELtemp{~}\ifx\TELtemp\TELfirst
125      \expandafter\telnumber\expandafter{\the\TELtoks}\TELtilde
126      \telprint{#2}%

```

```

127     \else
128         \TELtoks=\expandafter{\the\TELtoks#1}%
129         \TELsplit#2{}\TELsplitEND
130         \fi\fi\fi\fi\fi\fi\fi\fi
131     \fi
132 }

\telprint
133 \DeclareRobustCommand*\telprint}[1]{%
134     \TELtoks={}%
135     \TELsplit#1{}\TELsplitEND
136 }

137 \TELreset\let\TELreset=\UnDeFiNeD

```

3 Installation

3.1 Download

Package. This package is available on CTAN¹:

[CTAN:macros/latex/contrib/oberdiek/telprint.dtx](#) The source file.

[CTAN:macros/latex/contrib/oberdiek/telprint.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:macros/latex/contrib/oberdiek/oberdiek-tds.zip](#)

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
```

Script installation. Check the directory TDS:scripts/oberdiek/ for scripts that need further installation steps. Package attachfile2 comes with the Perl script pdfatfi.pl that should be installed in such a way that it can be called as pdfatfi. Example (linux):

```
chmod +x scripts/oberdiek/pdfatfi.pl
cp scripts/oberdiek/pdfatfi.pl /usr/local/bin/
```

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 telprint.dtx
```

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

```

telprint.sty → tex/generic/oberdiek/telprint.sty
telprint.pdf → doc/generic/oberdiek/telprint.pdf
telprint.dtx → source/generic/oberdiek/telprint.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.

¹<http://ftp.ctan.org/tex-archive/>

3.4 Refresh file name databases

If your \TeX distribution (te \TeX , mi \TeX , ...) relies on file name databases, you must refresh these. For example, te \TeX users run `texhash` or `mktextlsr`.

3.5 Some details for the interested

Attached source. The PDF documentation on CTAN also includes the `.dtx` source file. It can be extracted by AcrobatReader 6 or higher. Another option is `pdftk`, e.g. unpack the file into the current directory:

```
pdftk telprint.pdf unpack_files output .
```

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{telprint.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 pdf \LaTeX :

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

4 History

[1996/11/28 v1.0]

- Erste lauffähige Version.
- Nur `'` und `'/'` als zulässige Sonderzeichen.

[1997/09/16 v1.1]

- Dokumentation und Kommentare (Posting in de.comp.text.tex).
- Erweiterung um Sonderzeichen `'(,)', '+', '~'` und `' '`.
- Trennungsverhinderung am `'hyphen'`.

[1997/10/16 v1.2]

- Schutz vor wiederholtem Einlesen.
- Unter $\text{\LaTeX}_{2\epsilon}$ Nutzung des `\DeclareRobustCommand`-Features.

[1997/12/09 v1.3]

- Temporäre Variable eingespart.

[2004/11/02 v1.4]

- Fehler in der Dokumentation korrigiert.

[2005/09/30 v1.5]

- Konfigurierbare Symbole: '/', '(', ')', '+' und '~'.

[2006/02/12 v1.6]

- LPPL 1.3.
- Kurze Übersicht in Englisch.
- CTAN.

[2006/08/26 v1.7]

- New DTX framework.

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; numbers in *roman* refer to the code lines where the entry is used.

Symbols	L
\, 54	\leavevmode 56
Numbers	N
\0 70, 72	\newtoks 67
C	P
\csname 3, 13, 26, 39	\PackageInfo 18
	\ProvidesPackage 40
D	T
\DeclareRobustCommand . 45, 46, 47, 53, 55, 57, 59, 61, 63, 65, 91, 133	\TELdosplit 96, <u>97</u>
E	\TELfirst 105, 106, 109, 112, 115, 118, 121, 124
\empty 7, 72, 106	\TELfuture 96, 100, 101
\endcsname 3, 13, 26, 39	\TELhyphen 55, 110
\endinput 22, 43, 44	\telhyphen 2, <u>55</u>
F	\TELleftparen 59, 116
\futurelet 96	\telleftparen 2, <u>59</u>
H	\TELnumber 68, <u>94</u>
\hbox 56	\telnumber 2, 3, <u>91</u> , 102, 107, 110, 113, 116, 119, 122, 125
I	\TELnumberEND 68, 88, 94
\if 109, 112, 115, 118, 121	\TELplus 63, 122
\ifcase 4	\telplus 2, <u>63</u>
\ifnum 74, 81	\telprint 2, 2, 42, 103, 111, 114, 117, 120, 123, 126, <u>133</u>
\ifx 5, 7, 13, 26, 34, 42, 45, 72, 100, 106, 124	\TELreset 47, 51, 137
\immediate 15, 28, 48	\TELrightparen 61, 119
	\telrightparen 2, <u>61</u>
	\TELslash 57, 113

