The showexpl package

Rolf Niepraschk (Rolf.Niepraschk@gmx.de)

2014/01/19

1 Introduction

The documentation of a LATEX package is by far more readable if there are examples of the commands’ and environments’ usage. The best way to do that is to give a comparison of the LATEX code and the formatted output. showexpl is a package for doing that comparison, it is based on the package listings which provides a good typesetted source code with emphasised keywords and so on.

2 Usage

You can use showexpl like every other package by putting the line

\usepackage{showexpl}

in your source code. showexpl doesn’t know any options by itself, but all options for the underlying packages (listings and graphicx) will be passed to the respective packages.

showexpl provides one command and one environment:

- \LTXinputExample and
- \LTXexample

\LTXinputExample The syntax of \LTXinputExample is given by

\LTXinputExample[(key val list)]{file}

LTXexample The syntax of the environment \LTXexample is given by

\begin{LTXexample}[(key val list)]...\end{LTXexample}

The set of options represented by (key val list) is the same for both the command and the environment, the options are described in the following:

- **attachfile** Boolean valued key, default value: false. If set to true the sourcecode will be attached to the .pdf file—presumed that the document is processed by pdflatex.

- **codefile** Name of the (temporary) file that contains the code which will be formatted as source code. The default value is \jobname.tmp.

*This document corresponds to showexpl v0.3l, dated 2014/01/19.
explpreset A \texttt{\{key val list\}} which serves for presetting the properties of the formatting of the source code, for values see the documentation of the \texttt{listings} package. The default value is

\textbf{graphic} Name of a (graphic) file. This file—if present—will be included and displayed instead of the formatted code. The default value is empty.

\textbf{hsep} Defines the horizontal distance between the source code and the formatted text.

\textbf{justification} Defines the justification of the formatted text: reasonable values are \texttt{\textbackslash raggedleft}, \texttt{\textbackslash raggedright}, \texttt{\textbackslash centering}. The default value is \texttt{\textbackslash raggedright}.

\textbf{overhang} A \texttt{dimen}-value that defines the amount by which the formatted text and the source code can overlap the print space. The default value is 0 pt.

\textbf{pos:} Defines the relative position of the formatted text relating to the source code. Allowed values are t, b, l, r, o, and i for top, bottom, left, right, outer, and inner. The last values give sense only for two-sided printing, where there are outer and inner margins of a page. The default value is l.

\textbf{preset} Any \TeX{} code executed before the sample code but not visible in the listings area.

\textbf{rangeaccept} Boolean valued key, default value is false. If set to true, one can define ranges of lines that will be excerpted from the source code.

\textbf{rframe} Defines the form of the frame around the formatted text. With a non-empty value (e.g. “single”) a simple frame will be drawn. In the future more kinds of frames will be supported. The default value is empty (no frame).

\textbf{varwidth} Boolean valued key, default value is false. If set to true, the formatted text is set with its “natural” width instead of a fixed width as given by the value of the option \texttt{width}.

\textbf{hsep} Defines the vertical distance between the source code and the formatted text.

\textbf{wide} Boolean valued key, default value is false. If set to true, the source code and the formatted text overlap the print space and the margin area.

\textbf{width} A \texttt{\{dimen\}} value that defines the width of the formatted text. The default value depends of the relative positions of the source code and the formatted text.

\textbf{scaled} Without a value the formatted text will be scaled to fit the given width of the result area. With a number as value the formatted text will be scaled by this number.
3 Implementation

We must activate code from package listings for writing files.

\SX@defaultWD Parameter #2 is a length or a number. Parameter #1 is a macro. After a call of \SX@defaultWD this macro contains the value of the length or the value of the number multiplied by \linewidth.

Additional keys.

\lst@Key{pos} \def\SX@pos{#1}
\lst@Key{width} \def\SX@width{#1}
\lst@Key{hsep} \@tempdima=#1 \edef\SX@hsep{\the\@tempdima}
\lst@Key{vsep} \@tempdima=#1 \edef\SX@vsep{\the\@tempdima}
\lst@Key{overhang} \def\SX@overhang{#1}
\lst@Key{wide} \lstKV@SetIf{#1} \if\SX@wide
\lst@Key{rframe} \def\SX@rframe{#1}
\lst@Key{preset} \def\SX@preset{#1}
\lst@Key{scaled} \def\SX@scaled{#1}
\lst@Key{explpreset} \def\SX@scaled{#1}
\lst@Key{codefile} \def\SX@codefile{#1}
\if\SX@rangeaccept \lstKV@SetIf{#1} \if\SX@rangeaccept
\lst@Key{rangeaccept} \@tempdima=#1 \edef\SX@rangeacceptfalse{\the\@tempdima}
\lst@Key{varwidth} \@tempdima=#1 \edef\SX@varwidthfalse{\the\@tempdima}
\lst@Key{attachfile} \@tempdima=#1 \edef\SX@attachfilefalse{\the\@tempdima}
\if\SX@rangeaccept \lstKV@SetIf{#1} \if\SX@rangeaccept
\ld@Key{rangeaccept} \if\SX@rangeacceptfalse \@tempdima=#1 \edef\SX@rangeacceptfalse{\the\@tempdima}
\if\SX@varwidthfalse \@tempdima=#1 \edef\SX@varwidthfalse{\the\@tempdima}
\if\SX@attachfilefalse \@tempdima=#1 \edef\SX@attachfilefalse{\the\@tempdima}
\end{lstlisting}
\lst@Key{graphic}{\{}\% 
\lstKV@OptArg[width=\linewidth]{#1}{\%
    \edef\SX@graphicparam{##1}\edef\SX@graphicname{##2}\%
}\%
\% 
\newbox\SX@ResBox 
\newcommand*{\SX@pos}{}
\newcommand*{\SX@width}{}
\newcommand*{\SX@hsep}{}
\newcommand*{\SX@vsep}{}
\newcommand*{\SX@overhang}{}
\newcommand*{\SX@rframe}{}
\newcommand*{\SX@preset}{}
\newcommand*{\SX@explpreset}{}
\newcommand*{\SX@@explpreset}{}
\newcommand*{\SX@codefile}{\edef\SX@codefile{\jobname.tmp}}
\newcommand*{\SX@justification}{\raggedright}
\SX@@preset

\SX@@preset
Contains some redefinitions of \LaTeX{} macros and environments to do nothing. \SX@@preset will be called just before typesetting the result of the example code. More can be added with the user key “\texttt{\textasciitilde \texttt{\textasciitilde \textasciitilde}}\textasciitilde \texttt{\textasciitilde \textasciitilde}.”

\newcommand*{\SX@preset}{\% 
\renewcommand*{\documentclass}{\%} 
\renewcommand*{\usepackage}{\%} 
\renewenvironment*{document}{\%} 
\renewenvironment*{figure}{\%} 
\renewenvironment*{table}{\%} 
\renewcommand*{\cite}{\%} 
\let\tableofcontents\relax \let\listoffigures\relax \let\listoftables\relax \let\printindex\relax \let\listfiles\relax \let\nofiles\relax \let\index\@gobble \let\label\@gobble \let\immediate\relax \let\write\@gobbletwo \let\closeout\@gobble \let\@input\@gobble
\renewcommand*{\marginpar}{\%} 
\renewcommand*{\footnote}{\%} 
\let\@footnotetext\@gobble
\abovedisplayskip=\z@ \abovedisplayshortskip=\z@ 
\isSX@odd Parameter \#1 is executed on odd pages, parameter \#2 on even pages.
\newif\ifSX@wasodd \if@twoside 
\isSX@odd*{\isSX@odd}{\%} \ifthenelse{\isodd{\pageref{\SX@IDENT}}}{\%} \ifthenelse{\isodd{\pageref{\SX@IDENT}}}{\%} \else \fi 
\newcommand*{\SX@wasoddtrue}{\%} \newcommand*{\SX@wasoddfalse}{\%} \else \fi 
\f1
The call of \isSX@odd sets also \ifSX@wasodd to true or false. If it’s clear that no page break occurs, \ifSX@wasodd can be used.

\newcommand*{\SX@IDENT}{\SX@number\value\ltxexample}

\SX@attachfile

\newcommand*{\SX@attachfile}{\if@SX@attachfile
\attachfile[mimetype=text/plain,subject={example \theltxexample}]{\SX@codefile}\fi}

\SX@put@t/b/l/r/o/i
Six macros for positioning #2 (result) and #3 (code). The result can be above, below, left or right of the code area or on the outer or inner side. Parameter #1 is the width of the result.

\newcommand*{\SX@put@t[3]}{\SX@ResultArea{\linewidth}{#2}\endgraf\pagebreak[2]\setlength\@tempdima{\SX@vsep}\vskip\@tempdima\SX@CodeArea{\linewidth}{#3}}

\newcommand*{\SX@put@b[3]}{\SX@CodeArea{\linewidth}{#3}\endgraf\pagebreak[2]\setlength\@tempdima{\SX@vsep}\vskip\@tempdima\SX@ResultArea{\linewidth}{#2}}

\newcommand*{\SX@put@l[3]}{\setlength\@tempdimc{\linewidth-#1-\SX@hsep}\SX@ResultArea{#1}{#2}\hfill\SX@CodeArea{\@tempdimc}{#3}}

\newcommand*{\SX@put@r[3]}{\setlength\@tempdimc{\linewidth-#1-\SX@hsep}\SX@CodeArea{\@tempdimc}{#3}\hfill\SX@ResultArea{#1}{#2}}

\newcommand*{\SX@put@o[3]}{\@nameuse{SX@put@[ifSX@wasodd r\else l\fi}{#1}{#2}{#3}}

\newcommand*{\SX@put@i[3]}{\@nameuse{SX@put@[ifSX@wasodd l\else r\fi}{#1}{#2}{#3}}

\newcommand*{\SX@ResultArea[2]}{\SX@justification\setlength\@tempdima{#1}\%\minipage\@tempdima#2\endminipage\parbox{\@tempdima}{#2}}

\newcommand*{\SX@CodeArea[2]}{\setlength\@tempdima{#1}\%\minipage\@tempdima#2\endminipage\parbox{\@tempdima}{#2}}

\newcommand*{\SX@KillAboveCaptionskip}{\ifx\lst@caption\@empty\else\fi}

5
\lst@IfSubstring t\lst@captionpos
{\vskip-\abovecaptionskip}{}
\fi
\newcommand*\SX@KillBelowCaptionskip{%
\ifx\lst@caption\@empty
else
\lst@IfSubstring b\lst@captionpos
{\vskip-\belowcaptionskip}{}
\fi
}

\newenvironment{LTXexample}[1][]{%
@temptokena{#1}%
begingroup
For "codefile=..."/"graphic=..." if \thetxtexample or \thelstlisting is part of
the \filename.
advance\c@LTXexample\@ne advance\c@lstlisting\@ne
\edef\@tempa{\endgroup% Warum noetig?
\def\noexpand\SX@codefile{\SX@codefile}%
\def\noexpand\SX@graphicname{\SX@graphicname}%
\def\noexpand\SX@graphicparam{\SX@graphicparam}%.\edef\@tempa{%\endgroup% Warum noetig?
\let\lst@float=\relax\let\SX@float=\relax
\g@addto@macro\SX@@explpreset{,float=false}%
\edef\@tempa{\noexpand\lst@beginfloat{lstlisting}\[\SX@float]}%Without the following call \lst@beginfloat is undefined.
\@tempa%
\ifx\lst@caption\@empty
\advance\c@LTXexample\@ne \advance\c@lstlisting\@ne
\edef\@tempa{\endgroup%
\def\noexpand\SX@codefile{\SX@codefile}%
\def\noexpand\SX@graphicname{\SX@graphicname}%
\def\noexpand\SX@graphicparam{\SX@graphicparam}%.\edef\@tempa{%\endgroup% Warum noetig?
\let\lst@float=\relax\let\lst@float=\relax
\g@addto@macro\SX@@explpreset{,float=false}%
\edef\@tempa{\noexpand\lst@beginfloat{lstlisting}\[\SX@float]}%Without the following call \lst@beginfloat is undefined.
\@tempa%
\ifx\lst@caption\@empty
\advance\c@LTXexample\@ne \advance\c@lstlisting\@ne
\edef\@tempa{\endgroup%
\def\noexpand\SX@codefile{\SX@codefile}%
\def\noexpand\SX@graphicname{\SX@graphicname}%
\def\noexpand\SX@graphicparam{\SX@graphicparam}%.\edef\@tempa{%\endgroup% Warum noetig?
\let\lst@float=\relax\let\lst@float=\relax
\g@addto@macro\SX@@explpreset{,float=false}%
\edef\@tempa{\noexpand\lst@beginfloat{lstlisting}\[\SX@float]}%Without the following call \lst@beginfloat is undefined.
\@tempa%
\ifx\lst@caption\@empty
\advance\c@LTXexample\@ne \advance\c@lstlisting\@ne
\edef\@tempa{\endgroup%
\def\noexpand\SX@codefile{\SX@codefile}%
\def\noexpand\SX@graphicname{\SX@graphicname}%
\def\noexpand\SX@graphicparam{\SX@graphicparam}%.\edef\@tempa{%\endgroup% Warum noetig?
\let\lst@float=\relax\let\lst@float=\relax
\g@addto@macro\SX@@explpreset{,float=false}%
\edef\@tempa{\noexpand\lst@beginfloat{lstlisting}\[\SX@float]}%Without the following call \lst@beginfloat is undefined.
\@tempa%
\ifx\lst@caption\@empty
\advance\c@LTXexample\@ne \advance\c@lstlisting\@ne
\edef\@tempa{\endgroup%
\def\noexpand\SX@codefile{\SX@codefile}%
\def\noexpand\SX@graphicname{\SX@graphicname}%
\def\noexpand\SX@graphicparam{\SX@graphicparam}%.\edef\@tempa{%\endgroup% Warum noetig?
\let\lst@float=\relax\let\lst@float=\relax
\g@addto@macro\SX@@explpreset{,float=false}%
\edef\@tempa{\noexpand\lst@beginfloat{lstlisting}\[\SX@float]}%Without the following call \lst@beginfloat is undefined.
\@tempa%
Make $\texttt{\textbackslash SX@width}$ a real dimension if the unit is missing.

$\texttt{\textbackslash SX@defaultWD}\texttt{\textbackslash SX@width}\{\texttt{\textbackslash SX@width}\}$

Set the default width if necessary.

$\texttt{\textbackslash ifdim}\texttt{\textbackslash SX@width}<\texttt{\textbackslash z@}$

\begin{verbatim}
  \@tempswatrue
  \def\@tempa{t}
  \ifx\@tempa\texttt{\textbackslash SX@pos}\@tempswafalse\fi
  \def\@tempa{b}
  \ifx\@tempa\texttt{\textbackslash SX@pos}\@tempswafalse\fi
  \setlength\@tempdima{\linewidth+\texttt{\textbackslash SX@overhang}}
  \if\@tempswa\@tempdima=.5\@tempdima\fi
  \edef\texttt{\textbackslash SX@width}{\the\@tempdima}
\end{verbatim}

Correct $\texttt{\textbackslash SX@width}$ if a frame is requested.

$\texttt{\textbackslash ifx}\texttt{\textbackslash SX@frame}\texttt{\textbackslash @empty}$

\begin{verbatim}
  \long\def\texttt{\textbackslash SX@frame}\{\texttt{\textbackslash def}\texttt{\textbackslash \textbackslash SX@tempa}{}\}\texttt{\textbackslash def}\texttt{\textbackslash \textbackslash SX@tempa}{\texttt{\textbackslash fbox}}
  \setlength\@tempdima{\texttt{\textbackslash SX@width}-2\texttt{\textbackslash fboxsep}-2\texttt{\textbackslash fboxrule}}
  \edef\texttt{\textbackslash SX@width}{\the\@tempdima}
\end{verbatim}

Use the “natural” width of the result code if “varwidth” is true.

$\texttt{\textbackslash let}\texttt{\textbackslash SX@MakeCaption}\texttt{\textbackslash lst@MakeCaption}(t)\$%

$\texttt{\textbackslash setbox}\texttt{\textbackslash SX@ResBox}\texttt{\textbackslash hbox}\{\texttt{\textbackslash begin\textbackslash group}$

$\$}

$\$
Without a caption entry the command \lstinputlisting adds the filename to the “list of listings” (lol). This should be avoided.

The default parameters for all examples.

If "numbers=none" then margin dimensions should be zero.

All the default values.

Changing the defaults possible in showexpl.cfg.

Changing the defaults possible in showexpl.cfg.
Change History

v0.1a
General: “hpos” and “vpos” added, “pos” removed (RN). 3
Initial version .................. 1

v0.1b
\SX@put@t/b/l/r/o/i: Positioning the captions more independent of the result and code area (RN). ............... 5

v0.1c
\SX@put@t/b/l/r/o/i: Commands \SX@KillAboveCaptionskip and \SX@KillBelowCaptionskip added (RN). 5

v0.1f
General: “lstpreset” added. (RN). 3

v0.1h
General: “codefile” added. (RN) . 3
“lstpreset” renamed to “explpreset” (RN) . 3
New macro \LTInputExample (RN) . 10

LTExample: Renamed from “example” to “LTXexample” (RN) . 6

v0.1i
General: Better caption positioning and correct distance between the parts (RN) . 6

v0.1j
General: “rangeaccept” added (RN) . 3
\SX@input: For ranges of lines (RN) . 8

v0.1k
General: Some bug corrections (RN) . 3
\SX@put@t/b/l/r/o/i: Change \[a\]bove to \[t\]op (RN) . 5

v0.1l
General: “graphic” added (RN) . 3

v0.1m
General: Problem related to \label/\ref solved (RN) . 6

v0.2a
General: “varwidth” and “justifica-
tion” added (RN) . 3
“varwidth” package used (RN) . 6

v0.2b
General: Check if \SX@put@? is defined (RN) . 6

v0.3a
General: “attachfile” added (RN). 3
\SX@attachfile: Attach file functionality (with pdfTEx) added (RN) . 5

v0.3b
\SX@resultInput: Input of result code now inside a group; \makeatother added (RN) . 9

v0.3c
\SX@resultInput: Wrong catcode for newline char corrected (RN) . 9

v0.3d
\SX@resultInput: Missing \par added (RN) . 9
\SX@exclpreset: More redefinitions added (RN) . 4

v0.3g
General: \SX@ProcessResult is now working correctly using \readline and \scantokens. Thanks to Ulrich Diez for help (RN) . 8
Missing \newcommand for \SX@@explpreset added (RN) . 4

v0.3h
General: New Option ‘attachfiles’ (RN) . 3

v0.3j
\SX@put@code@result: Setting \lst@MakeCaption to was a bad idea for hyperlinks. Group added to varwidth environment. (Suggestions by Ulrike Fischer.) . 7

v0.3k
General: Definition for “hyperref” (suggested by Heiko Oberdiek) . 10
\SX@put@code@result: Setting \lst@MakeCaption to \@gobble again (prevent multiply defined labels; label key) . 7

v0.3l
General: Option “scaled” and \SX@scaled added (RN) . 3
\SX@resultInput: Code for “scaled” option (RN) . 9

11
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
\% .......................... 308, 309
\@input .................. 74
\@SX@attachfilefalse 36
\@SX@rangeacceptfalse 33
\@SX@varwidthfalse 34
\@SX@widefalse 35
\@addtofilelist ...... 203
\@captype ............... 64, 65
\@ehd ..................... 219, 270
\@firstofone .......... 321
\@footnotetext ....... 77
\@gobble .............. 70–72, 74, 77, 203, 208
\@gobbletwo ............ 73
\@inputcheck ....... 232, 234, 242, 254, 272, 283
\@latexerr ............ 269
\@temptokena .......... 147, 156
^ .................................. 280, 313

A
\abovecaptionskip ..... 136
\abovedisplayskip ..... 219
\arabic ................... 355
\attachfile ............ 94

B
\belowcaptionskip ..... 142
\bibliography .......... 71
\bigskipamount ....... 353
\box ....................... 221

C
\c@lstlisting .......... 149
\c@ltxexample ......... 149
\cite ..................... 66
\closein .............. 283
\closeout ............. 74
\columnsep .......... 353

E
\endgraf .......... 99, 104

environments:
\LTXexample ........ \REF{1, 145}
\LTXinputExample \REF{1, 145}

F
\@box .................. 196
\@boxrule ............ 197
\@boxsep .......... 268, 270
\@filename@area .... 270
\@filename@base .... 270
\@filename@ext .... 268, 270
\@filename@parse ... 267
\@footnote ......... 76

G
\@g@addtomacro ...... .172, 256, 341, 349
\@if@SX@attachfile .. 36, 40, 93
\@if@SX@rangeaccept .. 33, 37, 303, 315
\@if@SX@varwidth ...... 34, 38, 211, 216, 304
\@if@SX@wide .......... 26, 35, 179, 302
\@if@twoside .......... 83
\@ifeof ............... 232, 242
\@ifeof@Exists ........ 10, 265
\@if@SX@varodd ....... 82, 117, 120
\@if@thenelse ......... 85
\@includegraphics .... 332
\@index ............. 70
\@isodd ............ 85
\@isSX@odd ........ 82, 200

I
\@label ............... 70, 181
\@listoffigures ...... 67
\@listoftables ...... 68
\@lst@beginfloat .... 173
\@lst@BeginWriteFile 159
\@lst@belowskip ...... 206
\@lst@caption ....... 134, 140, 176
\@lst@captionpos .... 135, 141
\@lst@EndWriteFile \REF{1, 145}
\@lst@firstline .... \REF{233, 276, 297}
\@lst@float .... 168, 170, 171
\@lst@GetLineInterval \REF{250, 278}
\@lst@IfDisplaystyle 204
\@lst@IfSubString .... 135, 141
\@lst@Key ........... 21–28, 30–32, 37–40, 43
\@lst@lastline ..... \REF{246, 276, 299}
\@lst@lineno .... \REF{233, 235, 236, 246, 255, 258, 273}
\@lst@linestretch .... \REF{247, 273, 275, 276, 301}
\@lst@MakeCaption 205, 207, 208, 222, 223
\@lst@naglisting .. \REF{355}
\@lst@PlaceNumber .... 340
\@lst@RequireAspects 16
\@lst@setPreSet ...... \REF{273}
\@lst@tabluting .... \REF{344}
\@lst@KVoOptArg .... \REF{44}
\@lst@KVoSetff ...... \REF{26, 37, 38, 40}
\@lstnewenvironment \REF{145}
\@lstset .......... 150, 167, 169, 177, 274, 338, 339, 351
\@LTXexample (environ-

M
\@makeatother .......... 313
\@makebox .......... 201
\@MakePercentComment \REF{309, 313}
\@MakePercentIgnore \REF{308}
\@marginpar ....... 75
\@marginparsep .... 179
\@marginparwidth .... 179
\@newbox ............ 48

N
\@newbox ............ 48