

The `showexpl` package*

Rolf Niepraschk (Rolf.Niepraschk@gmx.de)

2020/05/26

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 $\langle key\ val\ list \rangle$ 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` v0.3r, dated 2020/05/26.

- explpreset** A *<key val list>* which serves for presetting the properties of the formatting of the source code, for values see the documentation of the `listings` package. The default value is
- 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.
- hsep** Defines the horizontal distance between the source code and the formatted text.
- justification** Defines the justification of the formatted text: reasonable values are `\raggedleft`, `\raggedright`, `\centering`. The default value is `\raggedright`.
- overhang** A *dimen*-value that defines the amount by which the formatted text and the source code can overlap the print space. The default value is 0pt.
- 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`.
- preset** Any \TeX code executed before the sample code but not visible in the listings area.
- 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.
- 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).
- 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 `width`.
- vsep** Defines the vertical distance between the source code and the formatted text.
- 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.
- width** A *<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.
- 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.

In addition to these options the kind of the result box (default: `\fbox`) can be changed. For example:

```
\renewcommand\ResultBox{\fcolorbox{green}{lightgray}}
\setlength\ResultBoxSep{5mm}% default: \fboxsep
\setlength\ResultBoxRule{2mm}% default: \fboxrule
```

3 Implementation

```

1 \DeclareOption{final}{%
2   \PassOptionsToPackage{\CurrentOption}{graphicx}%
3   \PassOptionsToPackage{\CurrentOption}{listings}%
4 }%
5 \DeclareOption{draft}{%
6   \PassOptionsToPackage{\CurrentOption}{graphicx}%
7   \PassOptionsToPackage{\CurrentOption}{listings}%
8 }%

9 \DeclareOption{attachfiles}{%
10  \AtBeginDocument{\IfFileExists{attachfile.sty}%
11    {\RequirePackage{attachfile}}{\def\SX@attachfile{}}}
12 }%
13 \DeclareOption*{\PassOptionsToPackage{\CurrentOption}{listings}}
14 \ProcessOptions\relax
15 \RequirePackage{refcount,listings,graphicx,varwidth,float}

```

We must activate code from package listings for writing files.

```

16 \lst@RequireAspects{writefile}

\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.

```

```

17 \newcommand*\SX@defaultWD[2]{%
18   \afterassignment\SX@def@WD\dimen@#2\linewidth\relax{#1}}
19 \newcommand*\SX@def@WD{}
20 \def\SX@def@WD#1\relax#2{\edef#2{\the\dimen@}}

```

Additional keys.

```

21 \lst@Key{pos}\relax{\def\SX@pos{#1}}
22 \lst@Key{width}\relax{\def\SX@width{#1}}
23 \lst@Key{hsep}\relax{@tempdima=#1\relax\edef\SX@hsep{\the\@tempdima}}
24 \lst@Key{vsep}\relax{@tempdima=#1\relax\edef\SX@vsep{\the\@tempdima}}
25 \lst@Key{overhang}\relax{\def\SX@overhang{#1}}
26 \lst@Key{wide}f[t]{\lstKV@SetIf{#1}\if@SX@wide}
27 \lst@Key{rframe}\relax{\def\SX@rframe{#1}}
28 \lst@Key{preset}\relax{\def\SX@preset{#1}}
29 \newcommand*\SX@scaled{}
30 \lst@Key{scaled}{?}[!]{\def\SX@scaled{#1}}

31 \lst@Key{explpreset}\relax{\def\SX@explpreset{#1}}
32 \lst@Key{codefile}\relax{\def\SX@codefile{#1}}
33 \newif\if@SX@rangeaccept \@SX@rangeacceptfalse
34 \newif\if@SX@varwidth \@SX@varwidthfalse
35 \newif\if@SX@wide \@SX@widefalse
36 \newif\if@SX@attachfile \@SX@attachfilefalse

37 \lst@Key{rangeaccept}f[t]{\lstKV@SetIf{#1}\if@SX@rangeaccept}

38 \lst@Key{varwidth}f[t]{\lstKV@SetIf{#1}\if@SX@varwidth}
39 \lst@Key{justification}\relax{\def\SX@justification{#1}}
40 \lst@Key{attachfile}f[t]{\lstKV@SetIf{#1}\if@SX@attachfile}
41 \newcommand*\SX@graphicname{}%
42 \newcommand*\SX@graphicparam{}%

```

```

43 \lst@Key{graphic}{ }[] {%
44   \lstKV@OptArg[width=\linewidth]{#1}{%
45     \edef\SX@graphicparam{##1}\edef\SX@graphicname{##2}%
46   }%
47 }%
48 \newbox\SX@ResBox
49 \newcommand\ResultBox{} \let\ResultBox=\fbox
50 \newdimen\ResultBoxSep \ResultBoxSep=\fboxsep
51 \newdimen\ResultBoxRule \ResultBoxRule=\fboxrule
52 \newcommand*\SX@pos{}
53 \newcommand*\SX@width{}
54 \newcommand*\SX@hsep{}
55 \newcommand*\SX@vsep{}
56 \newcommand*\SX@overhang{}
57 \newcommand*\SX@rframe{}
58 \newcommand\SX@preset{}
59 \newcommand*\SX@explpreset{}

60 \newcommand*\SX@@explpreset{}
61 \newcommand*\SX@codefile{}\edef\SX@codefile{\jobname.tmp}
62 \newcommand*\SX@justification{\raggedright}

```

\SX@@preset Contains some redefinitions of L^AT_EX 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 “**preset=...**”.

```

63 \newcommand*\SX@@preset{%
64   \renewcommand\documentclass[2] [] {\SX@eat@version}%
65   \renewcommand\usepackage[2] [] {\SX@eat@version}%
66   \renewenvironment{document}{}{}%
67   \renewcommand\cite[1] [] {}%
68   \let\tableofcontents\relax \let\listoffigures\relax
69   \let\listoftables\relax \let\printindex\relax
70   \let\listfiles\relax \let\nofiles\relax
71   \let\index\@gobble \let\label\@gobble
72   \let\refstepcounter=\@gobble
73   \let\bibliography\@gobble
74   \let\pagestyle\@gobble \let\thispagestyle\@gobble
75   %%\let\immediate\relax \let\write\@gobbletwo
76   %%\let\closeout\@gobble \let\@input\@gobble
77   \renewcommand\marginpar[2] [] {}%
78   \renewcommand\footnote[2] [] {}%
79   \let\@footnotetext\@gobble
80   %%\abovedisplayskip=\z@
81   %%\abovedisplayshortskip=\z@
82 }
83 \newcommand*\SX@eat@version[1] [] {}

```

\isSX@odd Parameter #1 is executed on odd pages, parameter #2 on even pages.

```

84 \newif\ifSX@wasodd
85 \if@twoside
86   \newcommand*\isSX@odd{%
87     \begingroup
88       \ifodd\getpagerefnumber{\SX@IDENT}%
89       \aftergroup\SX@wasoddttrue

```

```

90     \else
91         \aftergroup\SX@wasoddfalse
92     \fi
93 \endgroup
94 \ifSX@wasodd
95     \expandafter\@firstoftwo
96 \else
97     \expandafter\@secondoftwo
98 \fi
99 }
100 \else
101     \SX@wasodddtrue
102     \newcommand*\isSX@odd[2]{#1}
103 \fi

```

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.

```

104 \newcounter{ltxexample}
105 \newcommand*\SX@IDENT{\SX@number\value{ltxexample}}

```

`\SX@attachfile`

```

106 \newcommand*\SX@attachfile{%
107     \if@SX@attachfile
108         \attachfile[mimetype=text/plain,subject={example \theltxexample}]{%
109             {SX@codefile}}}%
110 \fi
111 }

```

`\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 innner side. Parameter #1 is the width of the result.

```

112 \newcommand*\SX@put@t[3]{%
113     \SX@ResultArea{\linewidth}{#2}\endgraf\pagebreak[2]%
114     \@tempdima=\dimexpr\SX@vsep\vskip\@tempdima
115     \SX@CodeArea{\linewidth}{#3}%
116 }
117 \newcommand*\SX@put@b[3]{%
118     \SX@CodeArea{\linewidth}{#3}\endgraf\pagebreak[2]%
119     \@tempdima=\dimexpr\SX@vsep\vskip\@tempdima
120     \SX@ResultArea{\linewidth}{#2}%
121 }
122 \newcommand*\SX@put@l[3]{%
123     \@tempdimc=\dimexpr\linewidth-#1-\SX@hsep %
124     \SX@ResultArea{#1}{#2}\hfill\SX@CodeArea{\@tempdimc}{#3}%
125 }
126 \newcommand*\SX@put@r[3]{%
127     \@tempdimc=\dimexpr\linewidth-#1-\SX@hsep %
128     \SX@CodeArea{\@tempdimc}{#3}\hfill\SX@ResultArea{#1}{#2}%
129 }
130 \newcommand*\SX@put@o[3]{%
131     \@nameuse{SX@put@\ifSX@wasodd r\else l\fi}{#1}{#2}{#3}%
132 }
133 \newcommand*\SX@put@i[3]{%
134     \@nameuse{SX@put@\ifSX@wasodd l\else r\fi}{#1}{#2}{#3}%

```

```

135 }
136 \newcommand\SX@ResultArea[2]{%
137   \SX@justification\@tempdima=\dimexpr #1 %
138   \parbox\@tempdima{#2}%
139 }
140 \newcommand\SX@CodeArea[2]{%
141   \@tempdima=\dimexpr #1 %
142   \sbox\@tempboxa{\parbox\@tempdima{#2}}%
143   \@tempdima=\dp\@tempboxa\usebox\@tempboxa
144   \rlap{\raisebox{-\@tempdima}[Opt][Opt]{\SX@attachfile}}%
145 }
146 \newcommand*\SX@KillAboveCaptionskip{%
147   \ifx\lst@caption\@empty\else
148     \lst@ifsubstring t\lst@captionpos
149       {\vskip-\abovecaptionskip}{}%
150   \fi
151 }
152 \newcommand*\SX@KillBelowCaptionskip{%
153   \ifx\lst@caption\@empty\else
154     \lst@ifsubstring b\lst@captionpos
155       {\vskip-\belowcaptionskip}{}%
156   \fi
157 }

```

LTExample

```

158 \lstnewenvironment{LTExample}[1][]{%
159   \@temptokena{#1}%
160   \begin{group}
161     For "codefile=..."/"graphic=..." if \thelstlisting is part of
162     the filename.
163     \advance\c@ltexample\@ne \advance\c@lstlisting\@ne
164     \expandafter\lstset\expandafter{\SX@explpreset,#1}%
165     \edef\x{\endgroup
166       \def\noexpand\SX@codefile{\SX@codefile}%
167       \def\noexpand\SX@graphicname{\SX@graphicname}%
168       \def\noexpand\SX@graphicparam{\SX@graphicparam}}%
169     \x
170     \xdef\SX@@explpreset{\the\@temptokena,codefile=\SX@codefile,%
171       graphic={[\SX@graphicparam]{\SX@graphicname}}}%
172     \setbox\@tempboxa=\hbox\bgroup
173     \lst@BeginWriteFile{\SX@codefile}%
174     \lst@EndWriteFile\egroup
175     \SX@put@code@result
176 }

```

\SX@put@code@result

```

177 \newcommand*\SX@put@code@result{%
178   \begin{group}
179     \expandafter\lstset\expandafter{\SX@explpreset}%
180     \expandafter\lstset\expandafter{\SX@@explpreset}%

```

Use listings floating procedure if necessary.

```

181 \ifx\lst@float\relax\else
182 \edef\@tempa{\noexpand\lst@beginfloat{lstlisting}[\lst@float]}
183 \expandafter\@tempa
184 \fi
185 \ifx\lst@caption\empty
186 \lstset{lolol=true}%
187 \fi
188 \if@SX@wide\def\SX@overhang{\marginparwidth+\marginparsep}\fi
189 \trivlist\item\relax
190 \stepcounter{ltexample}\label{SX@IDENT}%

```

Make \SX@width a real dimension if the unit is missing.

```

191 \SX@defaultWD\SX@width{\SX@width}%

```

Set the default width if necessary.

```

192 \ifdim\SX@width<\z@
193 \@tempswtrue
194 \def\@tempa{t}%
195 \ifx\@tempa\SX@pos\@tempswfalse\fi
196 \def\@tempa{b}%
197 \ifx\@tempa\SX@pos\@tempswfalse\fi
198 \@tempdima=\dimexpr\linewidth+\SX@overhang %
199 \if@tempswa\@tempdima=.5\@tempdima\fi%
200 \edef\SX@width{\the\@tempdima}%
201 \fi

```

Correct \SX@width if a frame is requested.

```

202 \ifx\SX@rframe\empty
203 \long\def\SX@frame##1{##1}%
204 \else
205 \let\SX@frame\ResultBox
206 \@tempdima=\dimexpr\SX@width-2\ResultBoxSep-2\ResultBoxRule %
207 \edef\SX@width{\the\@tempdima}%
208 \fi
209 \isSX@odd{\def\@tempa{l}}{\def\@tempa{r}}%
210 \makebox[\linewidth][\@tempa]{%
211 \parbox{\dimexpr\linewidth+\SX@overhang}{%

```

\SX@codefile (\jobname.tmp) is not necessary for the filelist.

```

212 \let\@addtofilelist\@gobble
213 \let\lst@ifdisplaystyle=\iftrue
214 \SX@KillAboveCaptionskip\lst@MakeCaption{t}%

```

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

```

215 \setbox\SX@ResBox\hbox{%
216 \fboxsep\ResultBoxSep
217 \fboxrule\ResultBoxRule
218 \SX@frame{%
219 \@nameuse{\if@SX@varwidth varwidth\else minipage}\fi}%
220 \SX@width\relax
221 \begingroup
222 \SX@resultInput
223 \endgroup
224 \@nameuse{end\if@SX@varwidth varwidth\else minipage}\fi}}%
225 \edef\SX@width{\the\wd\SX@ResBox}%

```

```

226         \@ifundefined{SX@put@SX@pos}%
227         {\@latex@error{Parameter ‘\SX@pos’ undefined}\@ehd}%
228         {\@nameuse{SX@put@SX@pos}%
229         {\SX@width}{\box\SX@ResBox}{\SX@codeInput}}}%
230         \lst@MakeCaption{b}\SX@KillBelowCaptionskip
231     }%
232 }%
233 \endtrivlist
234 \ifx\lst@float\relax\else\expandafter\lst@endfloat\fi
235 \gdef\SX@@explpreset{}%
236 \endgroup
237 }

```

```

238 \newcommand\SX@SkipToFirst{%
239     \ifeof\@inputcheck\else
240         \ifnum \lst@lineno=\lst@firstline\else
241             \readline\@inputcheck to\SX@tempa
242             \typeout{IGNORE (\the\lst@lineno)}%
243             \global\advance\lst@lineno\@ne
244             \SX@SkipToFirst
245         \fi
246     \fi
247 }
248 \newcommand\SX@ProcessResult{%
249     \ifeof\@inputcheck
250         \let\SX@tempb\relax
251     \else
252         \let\SX@tempb\SX@ProcessResult
253         \ifnum \lst@lineno>\lst@lastline\relax
254             \ifx\lst@linerange\@empty
255                 \let\SX@tempb\relax
256             \else
257                 \lst@GetLineInterval
258                 \SX@SkipToFirst
259             \fi
260         \else
261             \readline\@inputcheck to\SX@tempa
262             \typeout{READ (\the\lst@lineno)}%
263             \expandafter\g@addto@macro
264             \expandafter\SX@lines\expandafter{\SX@tempa^^J}%
265             \global\advance\lst@lineno\@ne
266         \fi
267     \fi
268     \SX@tempb
269 }

```

\SX@input

```

270 \newcommand\SX@input[1]{%
271     \begingroup
272     \IfFileExists{#1}{}%
273     {%
274         \filename@parse{#1}%
275         \ifx\filename@ext\relax \def\filename@ext{tex}\fi
276         \@latexerr{File

```



```

277     '\filename@area\filename@base.\filename@ext' not found.^^J^^J}\@ehd%
278 }%
279 \openin\@inputcheck#1
280 \lsthk@PreSet\let\lst@linrange\@empty\global\lst@lineno\@ne
281 \expandafter\lstset\expandafter{\SX@@explpreset}%
282 \ifx\lst@linrange\@empty
283     \edef\lst@linrange{\lst@firstline}-{\lst@lastline},}%
284 \fi
285 \lst@GetLineInterval
286 \SX@Info
287 \newlinechar='^^J\relax
288 \SX@SkipToFirst\let\SX@lines\@empty
289 \SX@ProcessResult
290 \closein\@inputcheck
291 \scantokens\expandafter{\SX@lines}%
292 \endgroup
293 }

294 \newcommand*\SX@Info{%
295     \typeout{-----}%
296     \typeout{pos=\SX@pos}%
297     \typeout{width=\SX@width}%
298     \typeout{hsep=\SX@hsep}%
299     \typeout{vsep=\SX@vsep}%
300     \typeout{overhang=\SX@overhang}%
301     \typeout{rframe=\SX@rframe}%
302     \typeout{codefile=\SX@codefile}%
303     \@ifundefined{lst@firstline}{}%
304     {\typeout{\string\lst@firstline=\lst@firstline}}%
305     \@ifundefined{lst@lastline}{}%
306     {\typeout{\string\lst@lastline=\lst@lastline}}%
307     \@ifundefined{lst@linrange}{}%
308     {\typeout{\string\lst@linrange=\lst@linrange}}%
309     \typeout{\string\if@SX@wide=\if@SX@wide TRUE\else FALSE\fi}%
310     \typeout{\string\if@SX@rangeaccept=\if@SX@rangeaccept TRUE\else FALSE\fi}%
311     \typeout{\string\if@SX@varwidth=\if@SX@varwidth TRUE\else FALSE\fi}%
312     \typeout{graphicfile=\SX@graphicname, graphicparameter=[\SX@graphicparam]}%
313     \typeout{-----}%
314 }
315 \providecommand*\MakePercentIgnore{\catcode'\%9\relax}
316 \providecommand*\MakePercentComment{\catcode'\%14\relax}

```

\SX@resultInput

```

317 \newcommand*\SX@resultInput{%
318     \ifx\SX@graphicname\@empty
319         \begingroup
320             \MakePercentComment\makeatother\catcode'\^M=5\relax
321             \SX@@preset\SX@preset
322             \if@SX@rangeaccept
323                 \let\SX@tempa=\SX@input
324             \else
325                 \let\SX@tempa=\input
326             \fi
327             \if\SX@scaled ?%

```

```

328     \let\SX@tempb=\@firstofone
329   \else
330     \if\SX@scaled !%
331       \def\SX@tempb##1{\resizebox{\SX@width}{!}{##1}}%
332     \else
333       \def\SX@tempb##1{\scalebox{\SX@scaled}{##1}}%
334     \fi
335   \fi
336   \let\SX@lst@Init=\lst@Init

```

Prevents float environments from floating. This is not enough for floating listing environments! Why?

```

337   \def\@xfloat##1[##2]{%
338     \def\@capttype{##1}%
339     \@namedef{the\@capttype}{0}%
340     \@float@HH{##1}[H]}%

```

Special handling of floating listing environments.

```

341   \def\lst@Init{%
342     \let\lst@float=\relax
343     \setcounter\@capttype{-1}%
344     \SX@lst@Init
345   }

```

Typeset the Code.

```

346   \SX@tempb{\SX@tempa{\SX@codefile}}\par
347   \endgroup
348   \else
349     \expandafter\includegraphics\expandafter[\SX@graphicparam]%
350     {\SX@graphicname}%
351   \fi
352 }

```

\SX@codeInput

```

353 \newcommand*\SX@codeInput{%
354   Without a caption entry the command \lstinputlisting adds the filename to
355   the “list of listings” (lol). This should be avoided.
356   \begingroup
357     The default parameters for all examples.
358     \expandafter\lstset\expandafter{\SX@explpreset}%
359     If ”numbers=none” then margin dimensions should be zero.
360     \expandafter\lstset\expandafter{\SX@@explpreset}%
361     \ifx\lst@PlaceNumber\@empty
362       \g@addto@macro\SX@@explpreset{xleftmargin=0pt,xrightmargin=0pt}%
363     \fi
364     \SX@Info
365     \expandafter\lstinputlisting\expandafter[\SX@@explpreset,nolol=true,%
366       caption={},belowskip=\z@,aboveskip=\z@,float=false]{\SX@codefile}%
367   \endgroup
368 }%

```

```

365 \newcommand*\LTxinputExample[2] [] {%
366   \g@addto@macro\SX@@explpreset{float=false,#1,codefile=#2}%
367   \SX@put@code@result}%

All the default values.
368 \lstset{explpreset={numbers=left,numberstyle=\tiny,numbersep=.3em,
Negative width means defaults.
369   xleftmargin=1em,columns=flexible,language=[LaTeX]TEX},pos=1,width=-99pt,
370   overhang=0pt,hsep=\columnsep,vsep=\bigskipamount,rframe=single}
.
371 \AtBeginDocument{%
372   \def\theHlstnumber{\theHlstlisting.\arabic{lstnumber}.\lst@neglisting}%
373 }

Changing the defaults possible in showexpl.cfg.
374 \InputIfFileExists{showexpl.cfg}{-}{-}

```

Change History

v0.1a	General: “rangeaccept” added (RN).	3
	General: “hpos” and “vpos” added, “pos” removed (RN). . .	3
	Initial version	1
v0.1b	\SX@put@t/b/l/r/o/i: Change [a]bove to [t]op (RN).	5
	\SX@put@t/b/l/r/o/i: Positioning the captions more independend 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 \LTxinputExample (RN).	11
	Renamed from “example” to “LTxexample” (RN).	6
v0.1i	General: Better caption positioning and correct distance between the parts (RN).	6
v0.1j	\SX@input: For ranges of lines (RN).	8
	General: “rangeaccept” added (RN).	3
	\SX@put@t/b/l/r/o/i: Change [a]bove to [t]op (RN).	5
	General: Some bug corrections (RN).	3
	General: “graphic” added (RN). . .	3
	General: Problem related to \label/\ref solved (RN). . . .	6
	General: “varwidth” and “justification” added (RN). . .	3
	“varwidth” package used (RN). . .	6
	General: Check if \SX@put@? is defined (RN).	6
	\SX@attachfile: Attach file functionality (with pdfTeX) added (RN).	5
	General: “attachfile” added (RN). .	3
	\SX@resultInput: Input of result code now inside a group; \makeatother added (RN). . . .	9
	\SX@resultInput: Wrong catcode for newline char corrected (RN).	9

v0.3d	\SX@resultInput: Missing \par added (RN).	9	v0.3l	\SX@resultInput: Code for “scaled” option (RN).	9
v0.3e	\SX@@preset: More redefinitions added (RN).	4		General: Option “scaled” and \SX@scaled added (RN).	3
v0.3g	General: \SX@ProcessResult is now working correctly using \readline and \scantokens. Thanks to Ulrich Diez for help (RN).	8	v0.3m	\SX@put@code@result: Wrong assignement for \lst@belowskip (RN).	7
	Missing \newcommand for \SX@@explpreset added (RN).	4	v0.3n	\SX@put@code@result: Use \ResultBox	7
v0.3h	General: New Option ‘attachfiles’ (RN).	3		General: Define \ResultBox etc.	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	v0.3p	\SX@@preset: Remove extra treatment of ‘figure’/‘table’ (RN).	4
v0.3k	\SX@put@code@result: Setting \lst@MakeCaption to \@gobble again (prevent multiply defined labels; label key)	7		\SX@put@code@result: Let’s leave \lst@MakeCaption untouched (RN).	7
	General: Definition for “hyperref” (suggested by Heiko Oberdiek)	11		\SX@resultInput: Better handling of floats (RN).	10
				\isSX@odd: Replace \isodd with \ifodd\getpagerefnumber (remove package ‘ifthen’) (RN).	4
				General: Remove package ‘calc’ (RN).	3
			v0.3q	\SX@resultInput: Floats should always be numbered 0 (RN).	10

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	\@footnotetext	78	\abovedisplayshortskip	
\%	314, 315		\abovedisplayskip	80
\@@input	75	. 71–73, 75, 78, 211	\abovedisplayskip	79
\@SX@attachfilefalse	36		\aftergroup	88, 90
\@SX@orangeacceptfalse	33	\@gobbletwo 74	\arabic	336, 373
		\@inputcheck 238, 240, 248, 260, 278, 289	\attachfile	107
\@SX@varwidthfalse	34			
\@SX@widefalse	35	\@latex@error 226		
		\@latexerr 275	B	
\@addtofilelist	211	\@secondoftwo 96	\belowcaptionskip	154
\@capttype	336, 338, 339, 343, 347	\@temptokena 158, 167	\bibliography	72
\@ehd	226, 276	\@xfloat 337	\bigskipamount	371
\@firstofone	327	\^ 286, 319	\box	228
\@firstoftwo	94	A	C	
\@float@HH	340	\abovecaptionskip 148	\c@lstlisting	160

<code>\c@ltxexample</code>	160	<code>\lst@BeginWriteFile</code> 170	<code>\newdimen</code>	50, 51
<code>\cite</code>	67	<code>\lst@caption</code>	<code>\newlinechar</code>	286
<code>\closein</code>	289	146, 152, 184	
<code>\closeout</code>	75	<code>\lst@captionpos</code> 147, 153	O	
<code>\columnsep</code>	371	<code>\lst@endfloat</code>	<code>\openin</code>	278
		<code>\lst@EndWriteFile</code> . 173		
D		<code>\lst@firstline</code>	P	
<code>\dimexpr</code> . . 113, 118,		<code>\pagebreak</code> . . . 112, 117	
122, 126, 136,		<code>\lst@float</code>	<code>\pagestyle</code>	73
140, 197, 205, 210		. 180, 181, 233, 342	<code>\printindex</code>	69
		<code>\lst@GetLineInterval</code>		
E		R	
<code>\endgraf</code>	112, 117	<code>\raggedright</code>	62
environments:		<code>\lst@ifdisplaystyle</code> 212	<code>\raisebox</code>	143
<code>LTXexample</code> . . . 1, <u>157</u>		<code>\lst@ifSubstring</code> . .	<code>\readline</code>	240, 260
		<code>\resizebox</code>	330
F		<code>\lst@Init</code>	<code>\ResultBox</code>	49, 204
<code>\fbox</code>	49	<code>\lst@Key</code>	<code>\ResultBoxRule</code>	
<code>\fboxrule</code>	51, 216	30–32, 37–40, 43	51, 205, 216
<code>\fboxsep</code>	50, 215	<code>\lst@lastline</code>	<code>\ResultBoxSep</code>	
<code>\filename@area</code>	276	50, 205, 215
<code>\filename@base</code>	276	<code>\lst@lineno</code>	<code>\rlap</code>	143
<code>\filename@ext</code> . 274, 276		. 239, 241, 242,		
<code>\filename@parse</code> . . . 273		252, 261, 264, 279	S	
<code>\footnote</code>	77	<code>\lst@linelrange</code> 253,	<code>\sbox</code>	141
		279, 281, 282, 307	<code>\scalebox</code>	332
G		<code>\lst@MakeCaption</code> . .	<code>\scantokens</code>	290
<code>\g@addto@macro</code>	<code>\setcounter</code> 339, 343, 347	
.	262, 359, 367	<code>\lst@neglisting</code> . . . 373	<code>\stepcounter</code>	189
<code>\getpagerefnumber</code> . 87		<code>\lst@PlaceNumber</code> . . 358	<code>\string</code> 303, 305, 307–310	
		<code>\lst@RequireAspects</code> 16	<code>\SX@@explpreset</code> . . .	
I		<code>\lsthk@PreSet</code>	60, 167,
<code>\if@SX@attachfile</code> .		<code>\lstinputlisting</code> . . 362	179, 234, 280,
.	36, 40, 106	<code>\lstKV@OptArg</code>	357, 359, 362, 367
<code>\if@SX@rangeaccept</code> .		<code>\lstKV@SetIf</code>	<code>\SX@@preset</code>	<u>63</u> , 320
.	33, 37, 309, 321	<code>\SX@attachfile</code>	
<code>\if@SX@varwidth</code> . . .		<code>\lstnewenvironment</code> . 157	11, <u>105</u> , 143
.	34, 38, 218, 223, 310	<code>\lstset</code>	<code>\SX@CodeArea</code> . . 114,	
<code>\if@SX@wide</code>	117, 123, 127, 139
.	26, 35, 187, 308	178, 179, 185,	<code>\SX@codefile</code> . 32, 61,	
<code>\if@twoside</code>	84	280, 356, 357, 369	108, 163, 167,
<code>\ifeof</code>	238, 248	<code>LTXexample</code> (environ-	170, 301, 346, 363
<code>\ifFileExists</code> . . 10, 271		ment)	<code>\SX@codeInput</code> . 228, <u>354</u>	
<code>\ifSX@wasodd</code>		<code>\LTXinputExample</code> 1, 366	18–20
.	83, 93, 130, 133	M	<code>\SX@defaultWD</code> . . <u>17</u> , 190	
<code>\immediate</code>	74	<code>\makeatother</code>	<code>\SX@eat@version</code> . . .	
<code>\includegraphics</code> . . 350		<code>\makebox</code>	64, 65, 82
<code>\index</code>	71	<code>\MakePercentComment</code>	<code>\SX@explpreset</code>	
<code>\isSX@odd</code>	<u>83</u> , 208	31, 59, 161, 178, 356
		<code>\SX@frame</code> . 202, 204, 217	
L		<code>\MakePercentIgnore</code> . 314	<code>\SX@graphicname</code> . . .	
<code>\label</code>	71, 189	<code>\marginpar</code>	41, 45, 164,
<code>\listoffigures</code> 68		<code>\marginparsep</code>	168, 311, 317, 351
<code>\listoftables</code>	69	<code>\marginparwidth</code> . . . 187		
<code>\lst@beginfloat</code> . . . 181		N		
		<code>\newbox</code>		

\SX@graphicparam ..	\SX@ProcessResult .	251, 254, 267,
..... 42, 45, 247, 251, 288	327, 330, 332, 346
165, 168, 311, 350	\SX@put@code@result	\SX@vsep
\SX@hsep 174, 176, 368	24, 55, 113, 118, 298
23, 54, 122, 126, 297	\SX@put@t	\SX@wasoddfalse ... 90
\SX@IDENT . 87, 104, 189	\SX@put@t/b/l/r/o/i <u>111</u>	\SX@wasoddtrue . 88, 100
\SX@Info .. 285, 293, 361	\SX@ResBox	\SX@width ... 22, 53,
\SX@input <u>269</u> , 322	. 48, 214, 224, 228	190, 191, 199,
\SX@justification .	\SX@ResultArea 112,	205, 206, 219,
..... 39, 62, 136	119, 123, 127, 135	224, 228, 296, 330
\SX@KillAboveCaptionskip	\SX@resultInput 221, <u>316</u>	
..... 145, 213	\SX@rframe	T
\SX@KillBelowCaptionskip	.. 27, 57, 201, 300	\theHlstnumber 373
..... 151, 229	\SX@save@cnt .. 336, 347	\thelstlisting 373
\SX@lines . 263, 287, 290	\SX@scaled	\theltxexample 107
\SX@lst@Init .. 335, 344	29, 30, 326, 329, 332	\thispagestyle 73
\SX@overhang . 25, 56,	\SX@SkipToFirst ...	U
187, 197, 210, 299	. 237, 243, 257, 287	\usebox
\SX@pos . 21, 52, 194,	\SX@tempa . 240, 260,	142
196, 225–227, 295	263, 322, 324, 346	W
\SX@preset . 28, 58, 320	\SX@tempb 249,	\write
		74