

The `showexpl` package*

Rolf Niepraschk (Rolf.Niepraschk@gmx.de)

2016/08/05

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` v0.3m, dated 2016/08/05.

- explpreset** A $\langle key\ val\ list \rangle$ 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 0 pt.
- 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`.
- hsep** 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.

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{listings,calc,ifthen,graphicx,varwidth}

```

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 \lstKey{graphic}{ } [] {%
44   \lstKV@OptArg[width=\linewidth]{#1}{%
45     \edef\SX@graphicparam{##1}\edef\SX@graphicname{##2}%
46   }%
47 }%
48 \newbox\SX@ResBox
49 \newcommand*\SX@pos{}
50 \newcommand*\SX@width{}
51 \newcommand*\SX@hsep{}
52 \newcommand*\SX@vsep{}
53 \newcommand*\SX@overhang{}
54 \newcommand*\SX@rframe{}
55 \newcommand\SX@preset{}
56 \newcommand*\SX@explpreset{}

57 \newcommand*\SX@@explpreset{}
58 \newcommand*\SX@codefile{}\edef\SX@codefile{\jobname.tmp}
59 \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=...**”.

```

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

```

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

```

82 \newif\ifSX@wasodd
83 \if@twoside
84   \newcommand*\isSX@odd[2] {%
85     \ifthenelse{\isodd{\pageref{\SX@IDENT}}}{%
86       {\SX@wasoddtrue #1}{\SX@wasoddfalse #2}}
87   \else
88     \newcommand*\isSX@odd[2] {\#1}\SX@wasoddtrue
89 \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.

```
90 \newcounter{ltexample}
91 \newcommand*{\SX@IDENT}{SX@number\value{ltexample}}
```

`\SX@attachfile`

```
92 \newcommand*\SX@attachfile{%
93   \ifSX@attachfile
94     \attachfile[mimetype=text/plain,subject={example \theltexample}]{%
95       {\SX@codefile}{}%
96   \fi
97 }
```

`\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.

```
98 \newcommand*\SX@put@t[3]{%
99   \SX@ResultArea{\linewidth}{#2}\endgraf\pagebreak[2]%
100  \setlength\@tempdima{\SX@vsep}\vskip\@tempdima
101  \SX@CodeArea{\linewidth}{#3}%
102 }
103 \newcommand*\SX@put@b[3]{%
104   \SX@CodeArea{\linewidth}{#3}\endgraf\pagebreak[2]%
105   \setlength\@tempdima{\SX@vsep}\vskip\@tempdima
106   \SX@ResultArea{\linewidth}{#2}%
107 }
108 \newcommand*\SX@put@l[3]{%
109   \setlength\@tempdimc{\linewidth-#1-\SX@hsep}%
110   \SX@ResultArea{#1}{#2}\hfill\SX@CodeArea{\@tempdimc}{#3}%
111 }
112 \newcommand*\SX@put@r[3]{%
113   \setlength\@tempdimc{\linewidth-#1-\SX@hsep}%
114   \SX@CodeArea{\@tempdimc}{#3}\hfill\SX@ResultArea{#1}{#2}%
115 }
116 \newcommand*\SX@put@o[3]{%
117   \@nameuse{SX@put@\ifSX@wasodd r\else l\fi}{#1}{#2}{#3}%
118 }
119 \newcommand*\SX@put@i[3]{%
120   \@nameuse{SX@put@\ifSX@wasodd l\else r\fi}{#1}{#2}{#3}%
121 }
122 \newcommand\SX@ResultArea[2]{%
123   \SX@justification\setlength\@tempdima{#1}%
124   %\minipage\@tempdima#2\endminipage
125   \parbox\@tempdima{#2}%
126 }
127 \newcommand\SX@CodeArea[2]{%
128   \setlength\@tempdima{#1}%
129   \sbox\@tempboxa{\parbox\@tempdima{#2}}%
130   \@tempdima=\dp\@tempboxa\usebox\@tempboxa
131   \rlap{\raisebox{-\@tempdima}[0pt][0pt]{\SX@attachfile}}%
132 }
133 \newcommand*\SX@KillAboveCaptionskip{%
134   \ifx\lst@caption\@empty\else
```

```

135 \lst@ifsubstring t\lst@captionpos
136 {\vskip-\abovecaptionskip}{}%
137 \fi
138 }
139 \newcommand*\SX@KillBelowCaptionskip{%
140 \ifx\lst@caption\@empty\else
141 \lst@ifsubstring b\lst@captionpos
142 {\vskip-\belowcaptionskip}{}%
143 \fi
144 }

```

LTXexample

```

145 \lstnewenvironment{LTXexample}[1] []
146 {%
147 \temptokena{#1}%
148 \beginngroup

```

For "codefile=..." / "graphic=..." if \theltxexample or \thelstlisting is part of the filename.

```

149 \advance\c@ltxexample\@ne \advance\c@lstlisting\@ne
150 \expandafter\lstset\expandafter{\SX@explpreset,#1}%
151 \edef\x{\endgroup
152 \def\noexpand\SX@codefile{\SX@codefile}%
153 \def\noexpand\SX@graphicname{\SX@graphicname}%
154 \def\noexpand\SX@graphicparam{\SX@graphicparam}}%
155 \x
156 \xdef\SX@@explpreset{\the\@temptokena,codefile=\SX@codefile,
157 graphic={[\SX@graphicparam]{\SX@graphicname}}}%
158 \setbox\@tempboxa=\hbox\bgroup% Warum noetig?
159 \lst@BeginWriteFile{\SX@codefile}%
160 }
161 {%
162 \lst@EndWriteFile\egroup
163 \SX@put@code@result
164 }

```

\SX@put@code@result

```

165 \newcommand*\SX@put@code@result{%
166 \beginngroup
167 \expandafter\lstset\expandafter{\SX@explpreset}%
168 \let\lst@float=\relax\let\SX@float=\relax

```

Without the following call \lst@beginfloat is undefined.

```

169 \expandafter\lstset\expandafter{\SX@@explpreset}%
170 \ifx\lst@float\relax\else

```

\lst@float must be \relax because the whole "example" should float but not the listings part in addition.

```

171 \let\SX@float=\lst@float\let\lst@float=\relax
172 \g@addto@macro\SX@@explpreset{,float=false}%
173 \edef\@tempa{\noexpand\lst@beginfloat{lstlisting}[\SX@float]}%
174 \expandafter\@tempa
175 \fi
176 \ifx\lst@caption\@empty

```

```

177     \lstset{ nolol=true}%
178     \fi
179     \if@SX@wide\def\SX@overhang{\marginparwidth+\marginparsep}\fi
180     \trivlist\item\relax
181     \stepcounter{ltxexample}\label{SX@IDENT}%
    Make \SX@width a real dimension if the unit is missing.
182     \SX@defaultWD\SX@width{\SX@width}%
    Set the default width if necessary.
183     \ifdim\SX@width<\z@
184     \@tempwattrue
185     \def\@tempa{t}%
186     \ifx\@tempa\SX@pos\@tempswafalse\fi
187     \def\@tempa{b}%
188     \ifx\@tempa\SX@pos\@tempswafalse\fi
189     \setlength\@tempdima{\linewidth+\SX@overhang}%
190     \if@tempswa\@tempdima=.5\@tempdima\fi%
191     \edef\SX@width{\the\@tempdima}%
192     \fi
    Correct \SX@width if a frame is requested.
193     \ifx\SX@rframe\@empty
194     \long\def\SX@frame##1{##1}%
195     \else
196     \let\SX@frame\fbbox
197     \setlength\@tempdima{\SX@width-2\fbboxsep-2\fbboxrule}%
198     \edef\SX@width{\the\@tempdima}%
199     \fi
200     \isSX@odd{\def\@tempa{l}}{\def\@tempa{r}}%
201     \makebox[\linewidth][\@tempa]{%
202     \parbox{\linewidth+\SX@overhang}{%
\SX@codefile (\jobname.tmp) is not necessary for the filelist.
203     \let\@addtofilelist\@gobble
204     \let\lst@ifdisplaystyle=\iftrue
205     \SX@KillAboveCaptionskip\lst@MakeCaption{t}%
206     \lst@belowskip=\z@
207     \let\lst@belowskip=\z@
    Use the “natural” width of the result code if “varwidth” is true. .
208     \let\SX@MakeCaption\lst@MakeCaption
209     \let\lst@MakeCaption\@gobble{}

210     \setbox\SX@ResBox\hbox{%
211     \SX@frame{%
212     \@nameuse{\if@SX@varwidth varwidth\else minipage\fi}%
213     \SX@width\relax
214     \begingroup
215     \SX@resultInput
216     \endgroup
217     \@nameuse{end\if@SX@varwidth varwidth\else minipage\fi}}%
218     \edef\SX@width{\the\wd\SX@ResBox}%
219     \@ifundefined{SX@put@SX@pos}%
220     {\@latex@error{Parameter ‘\SX@pos’ undefined}\@ehd}%
221     {\@nameuse{SX@put@SX@pos}}%

```

```

222         {\SX@width}{\box\SX@ResBox}{\SX@codeInput}}}%
223         \let\lst@MakeCaption\SX@MakeCaption
224         \lst@MakeCaption{b}\SX@KillBelowCaptionskip
225     }%
226 }%
227 \endtrivlist
228 \ifx\SX@float\relax\else\expandafter\lst@endfloat\fi
229 \gdef\SX@@explpreset{}%
230 \endgroup
231 }

```

```

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

```

\SX@input

```

264 \newcommand\SX@input[1]{%
265   \begingroup
266   \IfFileExists{#1}{}%
267   {%
268     \filename@parse{#1}%
269     \ifx\filename@ext\relax \def\filename@ext{tex}\fi
270     \@latexerr{File
271       ‘\filename@area\filename@base.\filename@ext’ not found.^^J^^J}\@ehd%
272   }%

```



```

273 \openin\@inputcheck#1
274 \lsthk@PreSet\let\lst@linerange\empty\global\lst@lineno\@ne
275 \expandafter\lstset\expandafter{\SX@@explpreset}%
276 \ifx\lst@linerange\empty
277 \edef\lst@linerange{\lst@firstline}-{\lst@lastline},}%
278 \fi
279 \lst@GetLineInterval
280 \SX@Info
281 \newlinechar='^^J\relax
282 \SX@SkipToFirst\let\SX@lines\empty
283 \SX@ProcessResult
284 \closein\@inputcheck
285 \scantokens\expandafter{\SX@lines}%
286 \endgroup
287 }

288 \newcommand*\SX@Info{%
289 \typeout{-----}%
290 \typeout{pos=\SX@pos}%
291 \typeout{width=\SX@width}%
292 \typeout{hsep=\SX@hsep}%
293 \typeout{vsep=\SX@vsep}%
294 \typeout{overhang=\SX@overhang}%
295 \typeout{rframe=\SX@rframe}%
296 \typeout{codefile=\SX@codefile}%
297 \ifundefined{lst@firstline}{}%
298 {\typeout{\string\lst@firstline=\lst@firstline}}%
299 \ifundefined{lst@lastline}{}%
300 {\typeout{\string\lst@lastline=\lst@lastline}}%
301 \ifundefined{lst@linerange}{}%
302 {\typeout{\string\lst@linerange=\lst@linerange}}%
303 \typeout{\string\if@SX@wide=\if@SX@wide TRUE\else FALSE\fi}%
304 \typeout{\string\if@SX@rangeaccept=\if@SX@rangeaccept TRUE\else FALSE\fi}%
305 \typeout{\string\if@SX@varwidth=\if@SX@varwidth TRUE\else FALSE\fi}%
306 \typeout{graphicfile=\SX@graphicname, graphicparameter=[\SX@graphicparam]}%
307 \typeout{-----}%
308 }

309 \providecommand*\MakePercentIgnore{\catcode'\%9\relax}
310 \providecommand*\MakePercentComment{\catcode'\%14\relax}

```

\SX@resultInput

```

311 \newcommand*\SX@resultInput{%
312 \ifx\SX@graphicname\empty
313 \begingroup
314 \MakePercentComment\makeatother\catcode'\^^M=5\relax
315 \SX@@preset\SX@preset
316 \if@SX@rangeaccept
317 \let\SX@tempa=\SX@input
318 \else
319 \let\SX@tempa=\input
320 \fi
321 \if\SX@scaled ?%
322 \let\SX@tempb=\@firstofone
323 \else

```

```

324     \if\SX@scaled !%
325     \def\SX@tempb##1{\resizebox{\SX@width}{!}{##1}}%
326     \else
327     \def\SX@tempb##1{\scalebox{\SX@scaled}{##1}}%
328     \fi
329     \fi
330     \SX@tempb{\SX@tempa{\SX@codefile}}\par
331 \endgroup
332 \else
333 \expandafter\includegraphics\expandafter[\SX@graphicparam]%
334 {\SX@graphicname}%
335 \fi
336 }

```

\SX@codeInput

```

337 \newcommand*\SX@codeInput{%
    Without a caption entry the command \lstinputlisting adds the filename to
    the “list of listings” (lol). This should be avoided.
338 \begingroup
    The default parameters for all examples.
339 \expandafter\lstset\expandafter{\SX@explpreset}%
    If “numbers=none” then margin dimensions should be zero.
340 \expandafter\lstset\expandafter{\SX@@explpreset}%
341 \ifx\lst@PlaceNumber\@empty
342 \g@addto@macro\SX@@explpreset{xleftmargin=0pt,xrightmargin=0pt}%
343 \fi
344 \SX@Info
345 \expandafter\lstinputlisting\expandafter%
346 [\SX@@explpreset,nolol=true,caption={}]{\SX@codefile}%
347 \endgroup
348 }%

349 \newcommand*\LTXinputExample[2][ ]{%
350 \g@addto@macro\SX@@explpreset{#1,codefile=#2}%
351 \SX@put@code@result}%

    All the default values.
352 \lstset{explpreset={numbers=left,numberstyle=\tiny,numbersep=.3em,
    Negative width means defaults.
353 xleftmargin=1em,columns=flexible,language=[LaTeX]TEX},pos=1,width=-99pt,
354 overhang=0pt,hsep=\columnsep,vsep=\bigskipamount,rframe=single}
.
355 \AtBeginDocument{%
356 \def\theHlstnumber{\thelstlisting.\arabic{lstnumber}.\lst@neglisting}%
357 }

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

```

Change History

v0.1a	General: “hpos” and “vpos” added, “pos” removed (RN). . . 3 Initial version 1	v0.3a	<code>\SX@attachfile</code> : Attach file functionality (with pdfTeX) added (RN). 5 General: “attachfile” added (RN). 3
v0.1b	<code>\SX@put@t/b/l/r/o/i</code> : Positioning the captions more independend of the result and code area (RN). 5	v0.3b	<code>\SX@resultInput</code> : Input of result code now inside a group; <code>\makeatother</code> added (RN). . . . 9
v0.1c	<code>\SX@put@t/b/l/r/o/i</code> : Commands <code>\SX@KillAboveCaptionskip</code> and <code>\SX@KillBelowCaptionskip</code> added (RN). 5	v0.3c	<code>\SX@resultInput</code> : Wrong catcode for newline char corrected (RN). 9
v0.1f	General: “lstpreset” added. (RN). 3	v0.3d	<code>\SX@resultInput</code> : Missing <code>\par</code> added (RN). 9
v0.1h	General: “codefile” added. (RN). . 3 “lstpreset” renamed to “explpreset” (RN). 3 <code>LTExample</code> : Renamed from “example” to “LTExample” (RN). 6	v0.3e	<code>\SX@@preset</code> : More redefinitions added (RN). 4
v0.1i	General: Better caption positioning and correct distance between the parts (RN). 6	v0.3g	General: <code>\SX@ProcessResult</code> is now working correctly using <code>\readline</code> and <code>\scantokens</code> . Thanks to Ulrich Diez for help (RN). 8 Missing <code>\newcommand</code> for <code>\SX@@explpreset</code> added (RN). 4
v0.1j	<code>\SX@input</code> : For ranges of lines (RN). 8 General: “rangeaccept” added (RN). 3	v0.3h	General: New Option ‘attachfiles’ (RN). 3
v0.1k	<code>\SX@put@t/b/l/r/o/i</code> : Change [a]bove to [t]op (RN). 5 General: Some bug corrections (RN). 3	v0.3j	<code>\SX@put@code@result</code> : Setting <code>\lst@MakeCaption</code> to was a bad idea for hyperlinks. Group added to varwidth environment. (Suggestions by Ulrike Fischer.). 7
v0.1l	General: “graphic” added (RN). . . 3	v0.3k	<code>\SX@put@code@result</code> : Setting <code>\lst@MakeCaption</code> to <code>\@gobble</code> again (prevent multiply defined labels; label key) 7
v0.1m	General: Problem related to <code>\label/\ref</code> solved (RN). . . . 6	v0.3l	<code>\SX@resultInput</code> : Code for “scaled” option (RN). 9 General: Option “scaled” and <code>\SX@scaled</code> added (RN). 3
v0.2a	General: “varwidth” and “justification” added (RN). . . . 3 “varwidth” package used (RN). . 6	v0.3m	<code>\SX@put@code@result</code> : Wrong assignement for <code>\lst@belowskip</code> (RN). 7
v0.2b	General: Check if <code>\SX@put@?</code> is defined (RN). 6		

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	environments:	
\% 309, 310	LTXexample ... <i>1</i> , <u>145</u>	\lst@EndWriteFile . 162
\@@input 74		\lst@firstline
\@SX@attachfilefalse 36	F 234, 277, 298
\@SX@rangeacceptfalse	\fbox 196	\lst@float 168, 170, 171
..... 33	\fboxrule 197	\lst@GetLineInterval
\@SX@varwidthfalse . 34	\fboxsep 197 251, 279
\@SX@widefalse 35	\filename@area 271	\lst@ifdisplaystyle 204
\@addtofilelist ... 203	\filename@base 271	\lst@ifsubstring ..
\@capttype 64, 65	\filename@ext . 269, 271 135, 141
\@ehd 220, 271	\filename@parse ... 268	\lst@Key 21–28,
\@firstofone 322	\footnote 76	30–32, 37–40, 43
\@footnotetext 77		\lst@lastline
\@gobble 70–	G 247, 277, 300
72, 74, 77, 203, 209	\g@addto@macro	\lst@lineno
\@gobbletwo 73	. 172, 257, 342, 350	. 234, 236, 237,
\@inputcheck 233, 235,	I	247, 256, 259, 274
243, 255, 273, 284	\if@SX@attachfile .	\lst@linelrange 248,
\@latex@error 220 36, 40, 93	274, 276, 277, 302
\@latexerr 270	\if@SX@rangeaccept .	\lst@MakeCaption 205,
\@temptokena .. 147, 156	.. 33, 37, 304, 316	208, 209, 223, 224
\^ 281, 314	\if@SX@varwidth ...	\lst@neglisting ... 356
	34, 38, 212, 217, 305	\lst@PlaceNumber .. 341
A	\if@SX@wide	\lst@RequireAspects 16
\abovecaptionskip . 136	.. 26, 35, 179, 303	\lsthk@PreSet 274
\abovedisplayshortskip	\if@twoside 83	\lstinputlisting .. 345
..... 79	\ifeof 233, 243	\lstKV@OptArg 44
\abovedisplayskip . 78	\IfFileExists .. 10, 266	\lstKV@SetIf
\arabic 356	\ifSX@wasodd 82, 117, 120 26, 37, 38, 40
\attachfile 94	\ifthenelse 85	\lstnewenvironment . 145
	\immediate 73	\lstset 150,
B	\includegraphics .. 333	167, 169, 177,
\belowcaptionskip . 142	\index 70	275, 339, 340, 352
\bibliography 71	\isodd 85	LTXexample (environ-
\bigskipamount 354	\isSX@odd <u>82</u> , 200	ment) <i>1</i> , <u>145</u>
\box 222		\LTXinputExample <i>1</i> , 349
	L	M
C	\label 70, 181	\makeatother 314
\c@lstlisting 149	\listoffigures 67	\makebox 201
\c@ltxexample 149	\listoftables 68	\MakePercentComment
\cite 66	\lst@beginfloat ... 173 310, 314
\closein 284	\lst@BeginWriteFile 159	\MakePercentIgnore . 309
\closeout 74	\lst@belowskip 206, 207	\marginpar 75
\columnsep 354	\lst@caption	\marginparsep 179
 134, 140, 176	\marginparwidth ... 179
E	\lst@captionpos 135, 141	N
\endgraf 99, 104	\lst@endfloat 228	\newbox 48

\newlinechar	281	\SX@defaultWD . .	17, 182	\SX@put@t/b/l/r/o/i	98
O					
\openin	273	\SX@eat@version	61, 62, 81	\SX@ResBox	48, 210, 218, 222
P					
\pagebreak	99, 104	\SX@explpreset	31, 56, 150, 167, 339	\SX@ResultArea . .	99, 106, 110, 114, 122
\pageref	85	\SX@float	168, 171, 173, 228	\SX@resultInput	215, 311
\pagestyle	72	\SX@frame . .	194, 196, 211	\SX@rframe	27, 54, 193, 295
\printindex	68	\SX@graphicname	41, 45, 153, 157, 306, 312, 334	\SX@scaled	29, 30, 321, 324, 327
R					
\raggedright	59	\SX@graphicparam	42, 45, 154, 157, 306, 333	\SX@SkipToFirst	232, 238, 252, 282
\raisebox	131	\SX@hsep	23, 51, 109, 113, 292	\SX@tempa . .	235, 255, 258, 317, 319, 330
\readline	235, 255	\SX@IDENT . .	85, 91, 181	\SX@tempb	244, 246, 249, 262, 322, 325, 327, 330
\resizebox	325	\SX@Info . .	280, 288, 344	\SX@vsep	24, 52, 100, 105, 293
\rlap	131	\SX@input	264, 317	\SX@wasoddfalse . . .	86
S					
\sbox	129	\SX@justification	39, 59, 123	\SX@wasoddtrue . .	86, 88
\scalebox	327	\SX@KillAboveCaptionskip	133, 205	\SX@width	22, 50, 182, 183, 191, 197, 198, 213, 218, 222, 291, 325
\scantokens	285	\SX@KillBelowCaptionskip	139, 224	T	
\stepcounter	181	\SX@lines . .	258, 282, 285	\theHlstnumber	356
\string 298, 300, 302–305		\SX@MakeCaption	208, 223	\thelstlisting	356
\SX@@explpreset	57, 156, 169, 172, 229, 275, 340, 342, 346, 350	\SX@overhang . .	25, 53, 179, 189, 202, 294	\theltxexample	94
\SX@@preset	60, 315	\SX@pos . .	21, 49, 186, 188, 219–221, 290	\thispagestyle	72
\SX@attachfile	11, 92, 131	\SX@preset . .	28, 55, 315	U	
\SX@CodeArea . .	101, 104, 110, 114, 127	\SX@ProcessResult	242, 246, 283	\usebox	130
\SX@codefile	32, 58, 95, 152, 156, 159, 296, 330, 346	\SX@put@code@result	163, 165, 351	W	
\SX@codeInput . .	222, 337	\SX@put@t	98	\write	73
\SX@def@WD	18–20				