The **leaflet** document class

Jürgen Schlegelmilch   Hubert Gäßlein   Rolf Niepraschk
Walter Schmidt

2004/12/22

Contents

1   Usage

Process the file `leaflet-manual.tex` with LaTeX to see details of usage and layout.

2   Implementation

1 ⟨∗class⟩

2.1   Saving some internal macros. . .

```
2 \let\LL@shipout\shipout \let\LL@outputpage\@outputpage
3 \let\LL@begindvi\@begindvi \let\LL@@end\@@end
```

2.2   Option processing

```
4 \@ifundefined{iflandscape}{\newif\iflandscape}{}%
5 \@ifundefined{iftumble}{\newif\iftumble}{}%
6 \newcommand\LL@debug@info[1]{}%
7 \DeclareOption{dvips}{\PassOptionsToPackage{\CurrentOption}{graphics}}
8 \DeclareOption{pdftex}{\PassOptionsToPackage{\CurrentOption}{graphics}}
9 \DeclareOption{vtex}{\PassOptionsToPackage{\CurrentOption}{graphics}}
10 \DeclareOption{dvipdfm}{\PassOptionsToPackage{\CurrentOption}{graphics}}
11 \DeclareOption{twoside}{\OptionNotUsed}
12 \DeclareOption{twocolumn}{\OptionNotUsed}
13 \DeclareOption{landscape}{\landscapetrue}
14 \DeclareOption{portrait}{\landscapefalse}
15 \DeclareOption{debug}{\let\LL@debug@info\typeout}
16 \DeclareOption{nospecialtricks}{%
17   \AtEndOfClass{%
18     \ifLL@combine
19       \let\immediate\@@@immediate\let\write\@@@write
20       \let\openout\@@@openout\let\closeout\@@@closeout
21       \let\special\@@@special\let\@@@exec@outs\relax
22     \fi}}
```

\LL@setPaperSize Some paper sizes are not supported by the article class. Hence this work-around. These page dimensions *must* be given in landscape orientation!

```
23 \newcommand*\LL@setPaperSize{}
24 \DeclareOption{a3paper}{\def\LL@setPaperSize{%
25   \paperwidth=420mm\paperheight=297mm\relax}}%
26 \@ifdefinable\ifLL@combine{\newif\ifLL@combine}
27 \DeclareOption{combine}{\LL@combinetrue}
28 \DeclareOption{nocombine}{\LL@combinefalse}
```

\LL@selectOutput

```
29 \newcommand*\LL@selectOutput{}
30 \DeclareOption{frontside}{\def\LL@selectOutput#1#2{#1}}
31 \DeclareOption{backside}{\def\LL@selectOutput#1#2{#2}}
32 \DeclareOption{bothsides}{\def\LL@selectOutput#1#2{#1#2}}

33 \DeclareOption{tumble}{\tumbletrue}
34 \DeclareOption{notumble}{\tumblefalse}
```

\LL@foldmark Drawing code for the fold mark. (Will be called within a `picture` environment.)

```
35 \newcommand*\LL@foldmark{}
36 \DeclareOption{foldmark}{%
37   \def\LL@foldmark{%
38     \begingroup
39       \linethickness{\LenToUnit{\foldmarkrule}}%
40       \setlength\@tempdima{\paperheight-\LL@tmargin}%
41       \put(0,\LenToUnit{\@tempdima}){%
42         \line(0,-1){\LenToUnit{\foldmarklength}}}%
43     \endgroup}%
44 }

45 \DeclareOption{nofoldmark}{\def\LL@foldmark{}}%

46 \newcommand*\LL@toomanypages[2]{}
47 \DeclareOption{draft}{\PassOptionsToClass{\CurrentOption}{article}%
48   \AtEndOfClass{%
49     \def\LL@toomanypages#1#2{%
50         \ClassWarningNoLine{leaflet}{#1.\MessageBreak#2}}%
51   }%
52 }
53 \DeclareOption{final}{\PassOptionsToClass{\CurrentOption}{article}%
54   \AtEndOfClass{%
55     \ifLL@combine
56       \def\LL@toomanypages#1#2{%
57         \ClassError{leaflet}{#1}{#2.}}%
58     \else
59       \def\LL@toomanypages#1#2{%
60         \ClassWarningNoLine{leaflet}{#1.\MessageBreak#2}}%
61     \fi
62   }%
63 }
64 \DeclareOption*{\PassOptionsToClass{\CurrentOption}{article}}
65 \PassOptionsToClass{landscape,a4paper}{article}
66 \ExecuteOptions{tumble,foldmark,bothsides,combine,landscape,final}
67 \ProcessOptions\relax
68 \ifLL@combine
```

\LL@rotate@I Rotates the output pages.
\LL@rotate@II
```
69   \newcommand*\LL@rotate@I{}\newcommand*\LL@rotate@II{}%
70   \iflandscape
71     \def\LL@rotate@I#1{#1}%
72     \iftumble
73       \def\LL@rotate@II#1{\rotatebox[origin=c]{180}{#1}}%
74     \else
75       \def\LL@rotate@II#1{#1}%
```

```
76        \fi
77     \else
78        \def\LL@rotate@I#1{\rotatebox[origin=c]{90}{#1}}%
79        \iftumble
80           \def\LL@rotate@II#1{\rotatebox[origin=c]{270}{#1}}%
81        \else
82           \def\LL@rotate@II#1{\rotatebox[origin=c]{90}{#1}}%
83        \fi
84     \fi
```

\immediate Im Folgenden definieren wir die TEX primitives \immediate, \write, \openout und \closeout
\write   um. Alle nicht "immediate" ausgeführten Anweisungen werden dem Makro \@@@pending@outs
\openout   (Kommaliste) hinzugefügt. So können speziell die \write-Anweisungen unabhängig von TEX's
\closeout   \shipout zu einem günstigen Zeitpunkt ausgeführt werden. For VTEX we must keep in mind
            that \immediate\special exist.

```
85     \def\@@@pending@outs{}\let\@@@immediate\immediate
86     \let\@@@write\write \let\@@@special\special
87     \let\@@@openout\openout \let\@@@closeout\closeout
88     \def\immediate{%
89       \let\write\immediate@write%
90       \let\openout\immediate@openout%
91       \let\closeout\immediate@closeout%
92       \let\special\immediate@special}%
93     \def\reset@immediate{%
94       \let\write\pending@write%
95       \let\openout\pending@openout%
96       \let\closeout\pending@closeout%
97       \let\special\@@@special}%
98     \long\def\pending@write#1#{\pending@@write{#1}}
99     \def\immediate@write{%
100      \reset@immediate\@@@immediate\@@@write}%
101    \def\immediate@openout{%
102      \reset@immediate\@@@immediate\@@@openout}%
103    \def\immediate@closeout{%
104      \reset@immediate\@@@immediate\@@@closeout}%
105    \def\immediate@special{%
106      \reset@immediate\@@@immediate\@@@special}%
107    \let\write\pending@write
108    \let\openout\pending@openout
109    \let\closeout\pending@closeout
```

\@dummy@whatsit Makes our redefined primitives more like "whatsits" (???). '\special{}' is also possible for
                 pdfTEX but we get a lot of strange warnings ("Non-PDF special ignored!") in this case.

```
110    \def\@dummy@whatsit{\special{}}
111    \begingroup\@ifundefined{pdfoutput}%
112    {\endgroup}
113    {\endgroup
114     \ifnum\pdfoutput>\z@\def\@dummy@whatsit{\pdfliteral{}}\fi}

115    \begingroup\expandafter\expandafter\expandafter\endgroup
116    \expandafter\ifx\csname eTeXversion\endcsname\relax
117      %%% Test is from Markus Kohm (d.c.t.t, 29 Jun 2004)
118      \ClassWarningNoLine{leaflet}{%
119      *********************************\MessageBreak
120      * It's very recommended to use eTeX  \MessageBreak
121      * with this package!                 \MessageBreak
122      *********************************}%
```

\pending@@write Doesn't work correctly without any \mark or \special or \pdfliteral command inside.

```
123      \long\def\pending@@write#1#2{%
124        \@dummy@whatsit
125        \g@addto@macro\@@@pending@outs{\@@@immediate\@@@write\number#1{#2},}}%
```

3

\pending@openout

```
126    \def\pending@openout#1 {%
127      \@dummy@whatsit
128      \g@addto@macro\@@@pending@outs{\@@@immediate\@@@openout\number#1,}}%
```

\pending@closeout

```
129    \def\pending@closeout#1{%
130      \@dummy@whatsit
131      \g@addto@macro\@@@pending@outs{\@@@immediate\@@@closeout\number#1,}}%
```

\@@@exec@outs Note to this implementation: The out commands and the page breaks are not allways in sync, which means possible wrong page references.

```
132      \newcommand*\@@@exec@outs{%
133        \@@@pending@outs\gdef\@@@pending@outs{}%
134        \LL@debug@info{%
135          >>> execute the output commands of the current page <<<}}%
```

```
136    \else
137    \RequirePackage{etex}
```

Allocation of a new mark class.

```
138      \globmarks\@@@out@mark
139      \newcounter{@@total@outs}\setcounter{@@total@outs}{0}
140      \newcounter{@@last@exec}\setcounter{@@last@exec}{0}
```

\pending@@write Doesn't work correctly without any \mark or \special or \pdfliteral command inside.

```
141    \long\def\pending@@write#1#2{%
142      \global\advance\c@@@total@outs\@ne%
143      \marks\@@@out@mark{\the\c@@@total@outs}%
144      \g@addto@macro\@@@pending@outs{\@@@immediate\@@@write\number#1{#2},}}%
```

\pending@openout
\pending@@openout
\pending@@@openout
\@ifequalsign

```
145 \def\pending@openout#1 {%
146 \global\advance\c@@@total@outs\@ne%
147 \marks\@@@out@mark{\the\c@@@total@outs}%
148 \g@addto@macro\@@@pending@outs{\@@@immediate\@@@openout\number#1,}}%
```

\pending@closeout

```
149 \def\pending@closeout#1{%
150 \global\advance\c@@@total@outs\@ne%
151 \marks\@@@out@mark{\the\c@@@total@outs}%
152 \g@addto@macro\@@@pending@outs{\@@@immediate\@@@closeout\number#1,}}%
```

\@@@exec@outs With the call of \@@@exec@outs we simulate TEX's execution of non-immediate output commands. The number of pending commands which can be executed must be decremented by \value{@@total@outs} − \botmarks\@@@out@mark. The remaining commands must still wait until the next output page is ready. It works only for output commands in the main text which moved to the next page but not for floating output commands.

```
153      \newcommand*\@@@exec@outs{%
154        \begingroup
```

Pending io's = \@tempcntb = TOTAL-OUTS − LAST-EXEC

```
155        \@tempcntb\c@@@total@outs\advance\@tempcntb-\c@@@last@exec%
156        \edef\reserved@a{\botmarks\@@@out@mark}%
157        \ifx\reserved@a\@empty\@tempcnta\z@\else\@tempcnta\reserved@a\fi%
158        \LL@debug@info{PENDING-OUTS:\the\@tempcntb\space\space
159                       TOTAL-OUTS:\the\c@@@total@outs\space\space
160                       LAST-EXEC:\the\c@@@last@exec\space\space
161                       TOPMARK:\topmarks\@@@out@mark\space\space
162                       FIRSTMARK:\firstmarks\@@@out@mark\space\space
163                       BOTMARK:\botmarks\@@@out@mark}%
```

4

Last valid pending entry = \@tempcntb = PENDING-OUTS − TOTAL-OUTS + BOTMARK
Number of invalid (moved) out's = \@tempcnta = TOTAL-OUTS − BOTMARK

```
164        \advance\@tempcnta-\c@@@total@outs \advance\@tempcntb\@tempcnta
165        \@tempcnta-\@tempcnta%
166        \ifnum\@tempcnta>\z@
167          \LL@debug@info{%
168            >>> resave \the\@tempcnta\space output command(s).
169                Too early to execute! <<<}%
170        \fi
171        \@tempcnta\z@ \def\reserved@b{}%
172        \@for\reserved@a :=\@@@pending@outs\do{%
173          \ifx\reserved@a\@empty\else
174            \ifnum\@tempcnta<\@tempcntb%
175              \reserved@a%  execute output's related to the current page box.
176              \global\advance\c@@@last@exec\@ne
177              \LL@debug@info{>>> execute output command number
178                  \the\c@@@last@exec\space<<<}%
179            \else
```

Other output's must be resaved and still pending.

```
180              \expandafter\g@addto@macro\expandafter\reserved@b\expandafter{%
181                \reserved@a,}%
182            \fi
183            \advance\@tempcnta\@ne%
184          \fi}%
185        \expandafter\@temptokena\expandafter{\reserved@b}%
186        \xdef\@@@pending@outs{\the\@temptokena}%
187      \endgroup}%

188    \fi% end of eTeX test.
```

\protected@write   Because our redefined \write is expandable we must protect it inside of LATEX's
                    \protected@write.

```
189    \long\def\protected@write#1#2#3{%
190      \begingroup
191        \let\thepage\relax
192        #2%
193        \let\protect\@unexpandable@protect
194        \edef\reserved@a{\noexpand\write#1{#3}}%
195        \reserved@a%
196      \endgroup
197      \if@nobreak\ifvmode\nobreak\fi\fi}%
```

\shipout   We make LATEX's \shipout inefficacious (no more shipout at all).

```
198    \def\shipout{\deadcycles\z@\setbox\@tempboxa=}
```

\@begindvi   Delay \@begindvi to be executed until the real page output.

```
199    \let\@begindvi\@empty
200 \fi% end of \ifLL@combine
```

\LL@toomanypages   Now elsewhere ...

```
201 \LoadClass{article}
202 \RequirePackage{everyshi,calc,graphicx}
```

\LL@pagesize@specials   Now the graphics driver is known. We will Check the graphics driver and set special code for
                        page dimensions and other.

```
203 \newcommand*\LL@pagesize@specials[2]{}
204 \@ifundefined{Gin@driver}{}%
205 {%
206   \ifx\Gin@driver\@empty\else%
207     \filename@parse{\Gin@driver}\@tempswafalse%
208     \def\reserved@a{dvips}%
```

```
209    \ifx\filename@base\reserved@a\@tempswatrue\fi%
210    \def\reserved@a{dvipdfm}%
211    \ifx\filename@base\reserved@a\@tempswatrue\fi%
212    \if@tempswa%
213      \ClassInfo{leaflet}{Generating code for dvips}%
214      \def\LL@pagesize@specials#1#2{%
215        \@tempdima=#1\@tempdimb=#2%
216        \AtBeginDvi{\special{papersize=\the\@tempdima,\the\@tempdimb}}}%
217    \fi%
218    \def\reserved@a{pdftex}%
219    \ifx\filename@base\reserved@a
220      \ClassInfo{leaflet}{Generating code for pdfTeX}%
221      \def\LL@pagesize@specials#1#2{%
222        \@tempdima=#1\@tempdimb=#2%
223        \pdfpagewidth\@tempdima\pdfpageheight\@tempdimb}%
224    \fi%
225    \def\reserved@a{vtex}%
226    \ifx\filename@base\reserved@a
227      \ClassInfo{leaflet}{Generating code for VTeX}%
228      \def\LL@pagesize@specials#1#2{%
229        \@tempdima=#1\@tempdimb=#2%
230        \mediawidth\@tempdima\mediaheight\@tempdimb}%
231    \fi%
232  \fi
233 }
```

\LL@CmdIgnored

```
234 \newcommand*\LL@CmdIgnored[1]{%
235   \ClassWarning{leaflet}{%
236     `\string#1' ignored}}
```

2.3   Changing the class properties

```
237 \setlength{\parskip}{1ex plus 2pt}
238 \@listi%
239 \setlength{\labelwidth}{\leftmargin}
240 \addtolength{\labelwidth}{-\labelsep}
241 \pagestyle{empty}
242 \headheight\z@
243 \headsep\z@
244 \footskip\z@
245 \marginparwidth\z@
246 \marginparsep\z@
247 \sloppy
248 \setcounter{secnumdepth}{0}
249 %\renewcommand\marginpar[2][]{\LL@CmdIgnored{\marginpar}}
250 \renewcommand\twocolumn[1][]{\LL@CmdIgnored{\twocolumn}}
251 \renewcommand\onecolumn{\LL@CmdIgnored{\onecolumn}}
252 \renewcommand\topfraction{0.7}
253 \renewcommand\bottomfraction{0.7}
254 \setlength{\textfloatsep}{10pt plus 4pt minus 3pt}
255 \setlength{\parindent}{\z@}
256 \setlength{\leftmargini}{1.5em}
257 \setlength{\leftmarginii}{1.5em}
258 \setlength{\leftmarginiii}{1.5em}
259 \setlength{\leftmarginiv}{1.5em}
260 \setlength{\leftmarginv}{1.5em}
261 \setlength{\leftmarginvi}{1.5em}
262 \setlength{\labelsep}{.5em}
263 \setlength  \labelwidth{\leftmargini}
264 \addtolength\labelwidth{-\labelsep}
```

\noparskip

```
265 \def\noparskip{\par\vspace{-\parskip}}
```

Modification of list and section parameters for a more compact layout.

```
266 \let\old@small\small
267 \renewcommand{\small}{\old@small\let\@listi\@listI}
268 \let\old@footnotesize\footnotesize
269 \renewcommand{\footnotesize}{\old@footnotesize\let\@listi\@listI}
```

\sectfont   User macro for changing the font of the typesetted parameter of the \?section commands.

```
270 \newcommand{\sectfont}{\bfseries}

271 \renewcommand\section{\@startsection{section}{1}{\z@}%
272    {-3.5ex \@plus -.75ex}%
273    {1ex} %{1.5ex}%
274    {\normalfont\large\sectfont}}
275 \renewcommand\subsection{\@startsection{subsection}{2}{\z@}%
276    {-2.5ex plus -.5ex}%
277    {1\p@} %{1ex}%
278    {\normalfont\normalsize\sectfont}}
279 \renewcommand\subsubsection{\@startsection{subsubsection}{3}{\z@}%
280    {-2.5ex plus -.5ex}%
281    {-1em}%
282    {\normalfont\normalsize\sectfont}}
283 \def\part{\LL@CmdIgnored{\part}\secdef\@part\@spart}
284 \def\@part[#1]#2{}
285 \def\@spart#1{}
286
287 \renewcommand*\descriptionlabel[1]{%
288    \hspace\labelsep\normalfont\descfont #1}
```

\descfont   User macro for changing the font of the description label.

```
289 \newcommand*\descfont{\bfseries}

290 \iffalse
291 % Doesn't work!
292 \g@addto@macro\enumerate{\parsep2\p@\@plus2\p@\@minus\z@}
293 \g@addto@macro\itemize{\parsep2\p@\@plus2\p@\@minus\z@}
294 \g@addto@macro\description{\parsep2\p@\@plus2\p@\@minus\z@}
295 \else
296 \newcommand*\LL@listsetup{%
297 %  \parsep.445ex\@plus.445ex\@minus.2225ex%
298 %  \parsep.5ex\@plus.5ex\@minus.25ex%
299    \parsep1ex\@plus.5ex\@minus.25ex%
300    \LL@debug@info{***parsep=\the\parsep}%
301 %  \itemsep.25\parsep
302    \itemsep\z@
303    \LL@debug@info{***itemsep=\the\itemsep}%
304    \topsep\z@
305    \LL@debug@info{***topsep=\the\topsep}%
306 %  \partopsep\z@
307    \LL@debug@info{***partopsep=\the\partopsep}%
308 }
309 \def\enumerate{%
310    \ifnum \@enumdepth >\thr@@\@toodeep\else
311      \advance\@enumdepth\@ne
312      \edef\@enumctr{enum\romannumeral\the\@enumdepth}%
313      \expandafter
314      \list
315        \csname label\@enumctr\endcsname
316        {\usecounter\@enumctr
317          \def\makelabel##1{\hss\llap{##1}}%
318          %\def\makelabel##1{##1\hfill}%
319          %\def\makelabel##1{\hss##1}%
320          \LL@listsetup
321        }%
322    \fi}
```

```
323 \def\itemize{%
324   \ifnum \@itemdepth >\thr@@\@toodeep\else
325     \advance\@itemdepth\@ne
326     \edef\@itemitem{labelitem\romannumeral\the\@itemdepth}%
327     \expandafter
328     \list
329       \csname\@itemitem\endcsname
330       {%
331         \def\makelabel##1{\hss\llap{##1}}%
332         %\def\makelabel##1{##1\hfill}%
333         %\def\makelabel##1{\hss##1}%
334         \LL@listsetup
335       }%
336   \fi}
337 \renewenvironment{description}
338               {\list{}{\labelwidth\z@ \itemindent-\leftmargin
339                     \let\makelabel\descriptionlabel
340                     \LL@listsetup}}
341               {\endlist}
342 \fi
```

2.4   Changing the page dimensions

\setmargins   User command to specify the margins. (o, u, l, r)
      \setmargins{⟨*top*⟩}{⟨*bottom*⟩}{⟨*left*⟩}{⟨*right*⟩}
      The new user interface! By default, the leaflet class does not use headings, footings and
marginal notes. If the user desires, these can be restored by setting the respective lengths *before*
calling \setmargins. Setting the margins should *always* be done using \setmargins; if the
user assigns to any of the length registers \topmargin, \evensidemargin, \oddsidemargin
using \setlength, chaos will happen.

```
343 \newcommand*\setmargins[4]{%
```

temp. top!

```
344    \setlength\topmargin{#1}%
345    \edef\LL@tmargin{\the\topmargin}%
```

temp. bottom!

```
346    \setlength\evensidemargin{#2}%
347    \setlength\textheight{%
348      \paperheight-\topmargin-\evensidemargin%
349      -\headheight-\headsep-\footskip}%
```

temp. left!

```
350    \setlength\oddsidemargin{#3}%
```

temp. right!

```
351    \setlength\evensidemargin{#4}%
352    \setlength\textwidth{%
353      \paperwidth-\oddsidemargin-\evensidemargin-\marginparwidth-\marginparsep}%
```

real top!

```
354    \addtolength\topmargin{-1in}%
```

real left!

```
355    \addtolength\oddsidemargin{-1in}%
```

probably not necessary

```
356    \evensidemargin\oddsidemargin%
357 }
```

The dimensions of a single small page.

```
358 \LL@setPaperSize
359 \paperwidth=0.333333334\paperwidth
360 \setmargins{11mm}{11mm}{8mm}{8mm}
```

2.5 Additional user macros

`\foldmarkrule`
`\foldmarklength` The dimension of the foldmark between page 2 and page 3.

```
361 \newcommand*\foldmarkrule{0.4pt}
362 \newcommand*\foldmarklength{2mm}
```

`\AddToBackground`
`\AddToBackground*` Adds the given parameter `#2` (some picture commands) to a macro which will be used inside a zero-dimensional picture environment located at the lower left corner of a page. Parameter `#1` selects the page number. The star version adds the picture commands to the background of one of the combined large pages instead of one of the small pages.

```
363 \newcommand\AddToBackground{%
364   \@ifstar{\@tempswatrue\LL@AddToBackground}
365         {\@tempswafalse\LL@AddToBackground}}
366 \@onlypreamble\AddToBackground
367 \newcommand\LL@AddToBackground[2]{%
368   \if@tempswa\def\@tempa{LL@largePic}\else\def\@tempa{LL@smallPic}\fi
369   \expandafter\providecommand\csname\@tempa\@Roman{#1}\endcsname{}%
370   \expandafter\g@addto@macro\csname\@tempa\@Roman{#1}\endcsname{#2}}
```

`\LenToUnit` `\LenToUnit{⟨dimension⟩}` can be used in a `picture` environment instead of multiples of `\unitlength`.

```
371 \newcommand\LenToUnit[1]{#1\@gobble}
```

`\CutLine`
`\CutLine*`
`\Scissors` Adds a dotted line with two scissors on the left side of the page given as parameter `#1`. The starred version draws only the dotted line.

```
372 \newcommand*\CutLine{%
373   \@ifstar{\@tempswatrue\LL@CutLine}{\@tempswafalse\LL@CutLine}}
374 \@onlypreamble\CutLine
375 \newcommand*\LL@CutLine[1]{%
376   \ifLL@combine
377     \ifx\Scissors\@empty\@tempswatrue\fi
378     \if@tempswa
379       \AddToBackground{#1}{%
380         \put(0,0){%
381           \rotatebox{90}{\makebox(\LenToUnit{\paperheight},0){%
382             \normalsize
383             \dotfill}}}}%
384     \else
385       \AddToBackground{#1}{%
386         \put(0,0){%
387           \rotatebox{90}{\makebox(\LenToUnit{\paperheight},0){%
388             \normalsize
389             \dotfill\Scissors\dotfill\dotfill\Scissors\dotfill}}}}%
390     \fi
391   \fi}
392 \IfFileExists{pifont.sty}
393   {\RequirePackage{pifont}%
394     \newcommand*\Scissors{\raisebox{-0.85ex}{\large\ding{34}}}}%
395   {\newcommand*\Scissors{}}
```

Adds the fold mark code between page 2 and page 3.

```
396 \AddToBackground{3}{\LL@foldmark}
```

2.6 Page construction and output

2.6.1 Intercepting the small pages

`\vb@xt@` Similar to `\hb@xt@` (should be part of the LaTeX kernel).

```
397 \providecommand*\vb@xt@{\vbox to}
```

We use our own page saving macro instead of real LaTeX output.

```
398 \AtBeginDocument{\EveryShipout{\LL@savePage}}
```

LL@page   A counter for the small pages (absolute counting).

399 `\newcounter{LL@page}\setcounter{LL@page}{1}`
400 `\newcommand\LL@tempa{}`

\LL@savePage   With "combine" option the six small pages will be saved in box registers (`\LL@boxI...\LL@boxVI`). The box255 with the page content must be a bit moved to get a "normal-sized" box.

401 `\newcommand*\LL@savePage{%`
402 `  \ifnum\c@LL@page<7\relax`

Add a background picture to a single small page.

403 `    \setbox\@cclv\vbox{%`
404 `      \vbox{\@tempdima=1in\relax`
405 `        \@tempdimb=\paperheight\advance\@tempdimb-\@tempdima`
406 `        \pictur@(0,0)(\LenToUnit{\@tempdima},\LenToUnit{\@tempdimb})%`
407 `          \begingroup`
408 `          \set@typeset@protect`
409 `          \@nameuse{LL@smallPic\Roman{LL@page}}%`
410 `          %\set@display@protect`
411 `          \endgroup`
412 `        \endpicture}%`
413 `        \nointerlineskip\box\@cclv}%`
414 `    \ifLL@combine`
415 `      \@@@exec@outs`
416 `      \expandafter\newsavebox\csname LL@box\Roman{LL@page}\endcsname%`
417 `      \setbox\@cclv=\vbox{\vskip1in\unvbox\@cclv}%`
418 `      \setbox\@cclv=\vbox{\moveright1in\box\@cclv}%`
419 `      \setbox\@cclv=\hb@xt@\paperwidth{\box\@cclv\hss}%`
420 `      \setbox\@cclv=\vb@xt@\paperheight{\box\@cclv\vss}%`
421 `      \global\expandafter\setbox%`
422 `        \csname LL@box\Roman{LL@page}\endcsname=\box\@cclv%`
423 `      \typeout{\@spaces[\the\c@LL@page] ==> [\Roman{LL@page}]}%`
424 `    \fi`
425 `  \fi`
426 `  \ifnum\c@LL@page=7\relax`

Make this an error or warning message, depending on class option. (`-->` warning in all cases?)

427 `    \begingroup`
428 `      \set@typeset@protect`
429 `      \LL@toomanypages{%`
430 `        The text you supplied fills more than six pages\MessageBreak`
431 `        and will therefore not fit onto a single flyer}{%`
432 `        Try using smaller fonts or reducing vertical space}%`
433 `    \endgroup`
434 `  \fi`
435 `  \stepcounter{LL@page}}`

2.6.2 Generating missing small pages

NOTE TO OURSELVES: All the following must be done if the document code is already processed.

436 `\ifLL@combine`

\@@end

437 `  \def\@@end{%`
438 `    \clearpage\pagestyle{empty}%`

Restore the original `\@outputpage` command (removing any related background pictures).

439 `    \let\@outputpage\LL@outputpage`

Suppresses all everyshi's clients (removing other kinds of background pictures) but our page saving macro is still needed for the next step and must be set again.

440 `    \def\@EveryShipout@Hook{}%`
441 `    \def\@EveryShipout@AtNextHook{}%`
442 `    \EveryShipout{\LL@savePage}%`

Create empty pages if necessary.

```
443    \loop\ifnum\c@LL@page<7\relax
444      \ClassInfo{leaflet}{Generating empty page \the\c@page}%
445      \null\newpage
446    \repeat
```

2.6.3 Reassembling the small pages

Here begins the real output using LaTeX's original `\shipout`. The everyshi package isn't active any more. The begin dvi hook is reactivated.

```
447    \let\shipout\LL@shipout \let\@begindvi\LL@begindvi
448    \paperwidth=3\paperwidth
449    \iflandscape
450      \LL@pagesize@specials{\paperwidth}{\paperheight}%
451    \else
452      \LL@pagesize@specials{\paperheight}{\paperwidth}%
453    \fi
```

\LL@shipoutPage   This is similar to LaTeX's `\@outputpage`, but without a group; `\set@display@protect` seems to work as well.

\LL@preparePages   Helper macros for page output: reassemble the small pages.

\LL@preparePage
```
454    \newcommand*\LL@shipoutPage[1]{%
455  %    \set@display@protect
456      \let \protect \noexpand
457      \shipout\vb@xt@\paperheight{%
458        \set@typeset@protect
459        \vskip-1in%
460        \@begindvi\hb@xt@\paperwidth{\hskip-1in##1\hss}\vss}}%
461    \newcommand*\LL@preparePages[3]{%
462      \typeout{[\@Roman{##1}\space\@Roman{##2}\space\@Roman{##3}] ==>}%
```

Add a background picture to a large page.

```
463      \pictur@(0,0)\@nameuse{LL@largePic\Roman{page}}\endpicture%
464      \LL@preparePage{##1}\LL@preparePage{##2}\LL@preparePage{##3}}%
465    \newcommand*\LL@preparePage[1]{%
466      \expandafter\box\csname LL@box\@Roman{##1}\endcsname}%
```

Combine and output the frontside and the backside.

```
467    \LL@selectOutput
468      {\setcounter{page}{1}%
469       \LL@shipoutPage{\LL@rotate@I{\LL@preparePages{5}{6}{1}}}}%
470      {\setcounter{page}{2}%
471       \LL@shipoutPage{\LL@rotate@II{\LL@preparePages{2}{3}{4}}}}%
472    \LL@@end
473  }%
```

End of the redefinition of `\@@end`.

```
474 \else
```

Create empty pages if necessary and then call the normal LaTeX end.

```
475  \LL@pagesize@specials{\paperwidth}{\paperheight}%
476  \AtEndDocument{%
477    \clearpage\pagestyle{empty}%
478    \loop\ifnum\c@LL@page<7\relax
479      \ClassInfo{leaflet}{Generating empty page \the\c@page}%
480      \null\newpage
481    \repeat
482  }
483 \fi
484 ⟨/class⟩
```

Done. Phew! (Did you hear that deep-drawn sigh?)

Change History

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.

13