

L^AT_EX-package for an easy declaration of functions and variables

Thomas Leineweber

1999/03/25

1 Overview

The package `method` can be used to easily format method- and variabledeclarations with L^AT_EX. It is based on work by J. Wahlmann and Robert Garmann.

2 Usage

The package is used as usual:

```
\usepackage [<language>]{method}
```

It defines two new environments: `method` and `data`. `Method` is used to typeset method-declarations, `data` for variable-declarations. At the moment the two options `english` and `german` are defined. With these options it is possible to select the language used to typeset the declarations. In the future some other languages will be added.

3 The environment `method`

`method` Within the environment `method` the following commands are defined:

- | | |
|------------------------|---|
| <code>\head</code> | • <code>\head{Header}</code> : The header of the method. |
| <code>\para</code> | • <code>\para{Name}{Description}</code> : Name and description of a parameter. |
| <code>\precond</code> | • <code>\precond{Precondition}</code> : Description of a precondition of the method. |
| <code>\descr</code> | • <code>\descr{Description}</code> : Description of the method itself. |
| <code>\postcond</code> | • <code>\postcond{postcondition}</code> : Description of a postcondition of the method. |
| <code>\error</code> | • <code>\error{Exception}</code> : Error and exceptions. |
| <code>\return</code> | • <code>\return{Return value}</code> : Description of the data returned by the method. |
| <code>\see</code> | • <code>\see{where}</code> : Cross-References. |

These commands have the following in common:

- All parameters are simple texts.
- The sequence of the commands inside the method-environment is not relevant. The parts are typeset automatically.
- Up to 26 `\para`-commands are allowed inside a method environment. When there are more, a warning will be issued and the following parameters will be ignored.
- The header of the method and the parameters are typeset in a typewriter font.
- If the header is extremely long, it can be typeset in more lines with the following macros (an example is given further down):

```

\headtabbed{<functionname>}      Name of the function
\headpara{<parametername>}      one or more parameter

```

4 The environment data

`data` The environment data equals to the environment method. The macros `\head`, `\descr` and `\see` can also be used inside a data-environment. Further macros inside a data-environment are:

- `\init` • `\init{Info}`: Information about the generation of the objekt.
- `\del` • `\del{Info}`: Information about the release of the object.

5 Examples

In this section some examples for the usage of the environments method and data are shown.

```

\begin{method}
\head{int div(int a, int b, double \&c);}
\para{a}{dividend}
\para{b}{divisor}
\para{\&c}{result of the division}
\precond{no preconditions}
\descr{Divides \texttt{a} by \texttt{b} and gives the result
in \texttt{c}}
\postcond{no postconditions}
\error{no errors}
\return{\texttt{-1}, when \texttt{b==0}, else \texttt{0}}
\see{your favourite mathematics book}
\end{method}

```

```

\begin{method}
\headtabbed{PrimObject()}
\headpara{const Matrix transformation,}
\headpara{AbstGeometry *geometry = 0,}
\headpara{MaterialApplication *material = 0,}
\headpara{AbstBumpMap *bumpMap = 0,}

```

```

        \headpara{Distribution *distribution = 0);}
    \para{transformation}{Transformation matrix}
    \para{*geometry}{\ldots}
    \descr{\ldots}
\end{method}

\begin{data}
    \head{char *name}
    \descr{Name of the user}
\end{data}
\begin{data}
    \head{char *no}
    \descr{Telephone-number of the user}
    \see{Telephone Book}
\end{data}

```

6 Identification und documentation

This package can only be used with L^AT_EX 2_ε. Therefore make sure, we use no other T_EX-format.

```

1 \NeedsTeXFormat{LaTeX2e}
    Show the name of the package and its version
2 \<+method>\ProvidesPackage{method}
3 \<+method>                [1999/03/25 v2.0b
4 \<+method>                LaTeX-package for method- and
5 \<+method>                data-descriptions (TL)]

```

We have a specialized class for the documentation.

```

6 \<*driver>
7 \documentclass[a4paper]{ltxdoc}

```

Set the specific options for the documentation of the package.

```

8 \DoNotIndex{" , \ , \addtolength, \begin, \CodelineIndex, \CodelineNumbered}
9 \DoNotIndex{\def, \DocInput, \documentclass, \DoNotIndex, \EnableCrossrefs}
10 \DoNotIndex{\end, \fbox, \fboxrule, \hfill, \hspace, \ifcase, \or, \fi}
11 \DoNotIndex{\ifnum, \fi, \item, \itemindent, \labelsep, \labelwidth}
12 \DoNotIndex{\leftmargin, \listparindent, \NeedsTeXFormat, \newcommand}
13 \DoNotIndex{\newcount, \newcounter, \newenvironment, \newlength, \sloppy}
14 \DoNotIndex{\nopagebreak, \PackageError, \parbox, \parindent, \stepcounter}
15 \DoNotIndex{\PrintChanges, \PrintIndex, \ProvidesPackage, \RecordChanges}
16 \DoNotIndex{\setcounter, \setlength, \textbf, \texttt, \usepackage, \vspace}
17 \DoNotIndex{\settowidth, \textwidth, \topsep}
18 \CodelineNumbered
19 \CodelineIndex
20 \EnableCrossrefs
21 \RecordChanges
22 \setcounter{StandardModuleDepth}{1}
23 \usepackage[T1]{fontenc}
24 \usepackage[latin1]{inputenc}

```

Give all details.

```

25 \begin{document}
26 \DocInput{method.dtx}

```

```

27 \PrintIndex
28 \PrintChanges
29 \end{document}
30 </driver>

```

7 Package internals

At the start of a method-environment the actual `textwidth` is read and saved for the layout of the description.

The commands for the parts, namely `\head`, `\para`, `\precond`, ..., define internal commands with the names `\meth@head`, `\meth@pa`, `\meth@pb`, ..., `\meth@pz`, `\meth@precond`, ... which are defined with the actual parameters.

At the end of a method-environment all these internal saved data is typeset in a (hopefully) fashionable way.

8 Helping commands

`\meth@paranum` The counter `\meth@paranum` counts the number of `\para`-commands within a method-environment:

```
31 \newcounter{meth@paranum}
```

The counter `\meth@headparanum` stores how many `\headpara`-commands are given within a method-environment:

```
32 \newcounter{meth@headparanum}
```

`\meth@totwid` The header will be typeset inside a framed minipage with a width of `\@totwid` (= `\textwidth` - 6mm):

```
33 \newlength{meth@totwid}
```

`\meth@indent` The descriptions are organized as lists with the following parameters:

```

\meth@listdecl 34 \def\meth@indent{3.5cm}
35 \def\meth@listdecl{\labelwidth3cm \labelsep0.5cm
36 \itemindent0cm \leftmargin\meth@indent
37 \topsep0cm \listparindent0cm}

```

`\meth@righttotwid` The right part of the list has a width of `\meth@righttotwid` (= `\meth@totwid` - `\meth@indent`):

```
38 \newlength{meth@righttotwid}
```

`\meth@namewid` The following lengths are used for the macro `\headtabbed`:

```

\meth@nameindent 39 \newlength{meth@namewid}
\meth@rightnamewid 40 \newlength{meth@nameindent}
41 \newlength{meth@rightnamewid}

```

9 Options

`\textsee` Now we have the option-processing. The option defines the language, which will be used to print the textual parts of the descriptions. At the moment only the languages english and german are defined.

```

\textreturn
\textprecond
\textpostcond
\textdescr
\texterror

```

First, define the parts for german descriptions

```
42 \DeclareOption{german}{\def\textsee{Siehe auch:}
43                     \def\textinit{Erzeugung:}
44                     \def\textdel{Freigabe:}
45                     \def\textreturn{R\ "uckgabewert:}
46                     \def\textprecond{Vorbed.:}
47                     \def\textpostcond{Nachbed.:}
48                     \def\textdescr{Beschreibung:}
49                     \def\texterror{Ausnahmebeh.:}}
50
```

Now for the english descriptions:

```
51 \DeclareOption{english}{\def\textsee{see also:}
52                     \def\textinit{initialisation:}
53                     \def\textdel{disposal:}
54                     \def\textreturn{return value:}
55                     \def\textprecond{precondition:}
56                     \def\textpostcond{postcondition:}
57                     \def\textdescr{description:}
58                     \def\texterror{exceptions:}}
```

The french descriptions, provided by Jean-Pierre Drucbert:

```
59 \DeclareOption{french}{\def\textsee{voir aussi:}
60                     \def\textinit{initialisation:}
61                     \def\textdel{lib\ 'eration:}
62                     \def\textreturn{valeur de retour:}
63                     \def\textprecond{pr\ 'econdition:}
64                     \def\textpostcond{postcondition:}
65                     \def\textdescr{description:}
66                     \def\texterror{exceptions:}}
```

Make the english version the default version and process the options.

```
67 \ExecuteOptions{english}
68 \ProcessOptions\relax
```

10 Error-detection

`\meth@where` The macro `\head` can be used both in the environments `method` and `data`. The
`\meth@checkdoubleopen` following value is used to differentiate if `method` or `data` is active. If none is active,
`\meth@checknotopen` the counter is set to 99. `Method` sets it to 0, `data` to 1.

```
69 \newcount\meth@where \meth@where=99
```

Now define the error messages:

```
70 \def\meth@checkdoubleopen{
71   \ifnum\meth@where<99
72     \PackageError{method}%
73       {There is an method.sty-environment open!}%
74     {}
75   \fi
76 }
77 \def\meth@checknotopen{
78   \ifnum\meth@where=99
79     \PackageError{method}%
80     {There is no method.sty-environment open!}%

```

```

81     {}
82   \fi
83 }

```

11 The environment method

`method` Now we define the environment method.

```

84 \newenvironment{method}
85 {

```

First we check, whether a method or data-environment is open. After that we set `\meth@where` to 0, which shows, that we are inside a method-environment.

```

86   \meth@checkdoubleopen
87   \meth@where=0

```

Define the lengths used for the typesetting of the method.

```

88   \setlength{\meth@totwid}{\textwidth}
89   \addtolength{\meth@totwid}{-6mm}
90   \setlength{\meth@righttotwid}{\meth@totwid}
91   \addtolength{\meth@righttotwid}{-\meth@indent}

```

The right column is not very wide. Therefore use `\sloppy`.

```

92   \sloppy

```

All parts are defined to nothing.

```

93   \def\meth@head{}
94   \def\meth@headtabbed{}
95   \setcounter{meth@headparanum}{0}
96   \def\meth@hpa{}\def\meth@hpb{}\def\meth@hpc{}\def\meth@hpd{}
97   \def\meth@hpe{}\def\meth@hpf{}\def\meth@hpg{}\def\meth@hph{}
98   \def\meth@hpi{}\def\meth@hpij{}\def\meth@hpk{}\def\meth@hpl{}
99   \def\meth@hpm{}\def\meth@hpn{}\def\meth@hpo{}\def\meth@hpp{}
100  \def\meth@hpnq{}\def\meth@hpr{}\def\meth@hps{}\def\meth@hpt{}
101  \def\meth@hpu{}\def\meth@hpuv{}\def\meth@hpw{}\def\meth@hpx{}
102  \def\meth@hpy{}\def\meth@hpz{}
103  \setcounter{meth@paranum}{0}
104  \def\meth@pa{}\def\meth@pb{}\def\meth@pc{}\def\meth@pd{}
105  \def\meth@pe{}\def\meth@pf{}\def\meth@pg{}\def\meth@ph{}
106  \def\meth@pi{}\def\meth@pj{}\def\meth@pk{}\def\meth@pl{}
107  \def\meth@pm{}\def\meth@pn{}\def\meth@po{}\def\meth@pp{}
108  \def\meth@pq{}\def\meth@pr{}\def\meth@ps{}\def\meth@pt{}
109  \def\meth@pu{}\def\meth@pv{}\def\meth@pw{}\def\meth@px{}
110  \def\meth@py{}\def\meth@pz{}
111  \def\meth@precond{}
112  \def\meth@descr{}
113  \def\meth@postcond{}
114  \def\meth@error{}
115  \def\meth@return{}
116  \def\meth@see{}

```

Now for the end of the environment. The first line in a paragraph is not indented and a small space is made above the header.

```

117 }{
118   \parindent0cm
119   \vspace{2mm}

```

now a sorted list of all given parts inside the environment. The user has to see, that only one of `\meth@head` and `\meth@headtabbed` is used.

```

120 \meth@head\meth@headtabbed
121 \nopagebreak[4]
122 \meth@pa \meth@pb \meth@pc \meth@pd \meth@pe \meth@pf \meth@pg
123 \meth@ph \meth@pi \meth@pj \meth@pk \meth@pl \meth@pm \meth@pn
124 \meth@po \meth@pp \meth@pq \meth@pr \meth@ps \meth@pt \meth@pu
125 \meth@pv \meth@pw \meth@px \meth@py \meth@pz
126 \meth@precond
127 \meth@descr
128 \meth@postcond
129 \meth@error
130 \meth@return
131 \meth@see

```

Now set `\meth@where` back to 99.

```
132 \meth@where=99
```

This is the end of the definition of the environment method.

```
133 }
```

12 The environment data

`data` The environment data is nearly equivalent to the environment method.

```

134 \newenvironment{data}
135 {
136 \meth@checkdoubleopen
137 \meth@where=1
138 \setlength{\meth@totwid}{\textwidth}
139 \addtolength{\meth@totwid}{-6mm}
140 \sloppy
141 \def\meth@head{}
142 \def\meth@descr{}
143 \def\meth@init{}
144 \def\meth@del{}
145 \def\meth@see{}
146 }{
147 \parindent0cm
148 \vspace{2mm}
149 \meth@head
150 \nopagebreak
151 \meth@descr
152 \meth@init
153 \meth@del
154 \meth@see
155 \meth@where=99
156 }

```

13 Macros for the parts inside the environments

The definitions for the parts of the environments data and method

13.1 `\head`

`\head` The macro `\head` is used to typeset the header of the method or the definition of the variable.

```
157 \newcommand{\head}[1]{
  First check, if the environment is active at the moment.
158   \meth@checknotopen
  If \meth@where is set to 0, the environment method is active.
159   \ifnum\meth@where=0
160     \def\meth@head{
  The code to typeset the header.
161       {\setlength{\fboxrule}{0.2mm}%
162        \fbox{\parbox{\meth@indent}{\hfill}
163              \begin{minipage}{\meth@righttotwid}
164                {\parindent-\meth@indent \texttt{#1}}
165              \end{minipage}
166             }}
167     }
168   \fi%
  If \meth@where is set to 1, the environment data is active.
169   \ifnum\meth@where=1
170     \def\meth@head{
  The code to typeset the header.
171       {\setlength{\fboxrule}{0.1mm}%
172        \fbox{\hspace{2mm}\begin{minipage}{\meth@totwid}
173              \texttt{#1}
174            \end{minipage}}
175     }
176   }
177   \fi%
178 }
```

13.2 `\headtabbed`

`\headtabbed` `\headtabbed` takes care of the first line, which will be formatted in the header of the method and defines the macro `\meth@headtabbed`, which does the output.

```
179 \newcommand{\headtabbed}[1]{
180   \meth@checknotopen
181   \def\meth@headtabbed{
182     \setlength{\fboxrule}{0.2mm}%
183     \settowidth{\meth@namewid}{\texttt{#1}}%
184     \setlength{\meth@rightnamewid}{\meth@totwid}
185     \addtolength{\meth@rightnamewid}{-\meth@namewid}
186     \setlength{\meth@nameindent}{\meth@namewid}
187     \addtolength{\meth@nameindent}{2mm}
188     \fbox{\parbox{\meth@nameindent}{\hfill}%
189           \begin{minipage}{\meth@rightnamewid}
190             \parindent-\meth@namewid
191             \texttt{#1}\meth@hpa
192           \meth@hpb
```



```

193         \meth@hpc
194         \meth@hpd
195         \meth@hpe
196         \meth@hpf
197         \meth@hpg
198         \meth@hph
199         \meth@hpi
200         \meth@hpi
201         \meth@hpk
202         \meth@hpl
203         \meth@hpm
204         \meth@hpn
205         \meth@hpo
206         \meth@hpp
207         \meth@hpq
208         \meth@hpr
209         \meth@hps
210         \meth@hpt
211         \meth@hpu
212         \meth@hqv
213         \meth@hqw
214         \meth@hpx
215         \meth@hpy
216         \meth@hpy}%
217     \end{minipage}%
218 }
219 }
220 }

```

13.3 \headpara

`\meth@defheadpara` `\meth@defheadpara` searches for the first empty macro of `\meth@hpa`, ..., `\meth@hpy`. In this macro the new line can be saved. This macro is used by `\headpara`.

```

221 \newcommand{\meth@defheadpara}[1]{
222   \ifcase\value{meth@headparanum}
223     \def\meth@hpa{#1} \or
224     \def\meth@hpb{#1} \or
225     \def\meth@hpc{#1} \or
226     \def\meth@hpd{#1} \or
227     \def\meth@hpe{#1} \or
228     \def\meth@hpf{#1} \or
229     \def\meth@hpg{#1} \or
230     \def\meth@hph{#1} \or
231     \def\meth@hpi{#1} \or
232     \def\meth@hpi{#1} \or
233     \def\meth@hpk{#1} \or
234     \def\meth@hpl{#1} \or
235     \def\meth@hpm{#1} \or
236     \def\meth@hpn{#1} \or
237     \def\meth@hpo{#1} \or
238     \def\meth@hpp{#1} \or
239     \def\meth@hpq{#1} \or

```

```

240 \def\meth@hpr{\#1} \or
241 \def\meth@hps{\#1} \or
242 \def\meth@hpt{\#1} \or
243 \def\meth@hpu{\#1} \or
244 \def\meth@hqv{\#1} \or
245 \def\meth@hqw{\#1} \or
246 \def\meth@hpx{\#1} \or
247 \def\meth@hpy{\#1} \or
248 \def\meth@hpz{\#1} \or
249 \PackageError{method}%
250 {Too many parameters in method-environment !}{ }
251 \fi
252 \stepcounter{meth@headparanum}
253 }

```

`\meth@defheadpara` Now for the definition of the macro `\headpara`. Check first, if the correspondent environment is open. Then use the macro `\meth@defheadpara`.

```

254 \newcommand{\headpara}[1]{
255 \meth@checknotopen
256 \meth@defheadpara{#1}
257 }

```

13.4 `\para`

`\meth@defpara`

```

258 \newcommand{\meth@defpara}[1]{
259 \ifcase\value{meth@paranum}
260 \def\meth@pa{#1} \or
261 \def\meth@pb{#1} \or
262 \def\meth@pc{#1} \or
263 \def\meth@pd{#1} \or
264 \def\meth@pe{#1} \or
265 \def\meth@pf{#1} \or
266 \def\meth@pg{#1} \or
267 \def\meth@ph{#1} \or
268 \def\meth@pi{#1} \or
269 \def\meth@pj{#1} \or
270 \def\meth@pk{#1} \or
271 \def\meth@pl{#1} \or
272 \def\meth@pm{#1} \or
273 \def\meth@pn{#1} \or
274 \def\meth@po{#1} \or
275 \def\meth@pp{#1} \or
276 \def\meth@pq{#1} \or
277 \def\meth@pr{#1} \or
278 \def\meth@ps{#1} \or
279 \def\meth@pt{#1} \or
280 \def\meth@pu{#1} \or
281 \def\meth@pv{#1} \or
282 \def\meth@pw{#1} \or
283 \def\meth@px{#1} \or
284 \def\meth@py{#1} \or
285 \def\meth@pz{#1} \or

```

```

286 \PackageError{method}%
287 {Too many parameters in method.sty-environment !}
288 \fi
289 \stepcounter{meth@paranum}
290 }

```

`\para` Here the definition for `\para`:

```

291 \newcommand{\para}[2]{
292 \meth@checknotopen
293 \meth@defpara{\begin{list}{\texttt{#1}}{\meth@listdecl}
294 \item #2
295 \end{list}}
296 }

```

13.5 The other macros

`\precond` The other macros are very simple. They create a list-environment and put their data inside of this list.

```

297 \newcommand{\precond}[1]{
298 \meth@checknotopen
299 \def\meth@precond{\begin{list}{\textbf{\textprecond}}{\meth@listdecl}
300 \item #1
301 \end{list}}
302 }

```

`\postcond`

```

303 \newcommand{\postcond}[1]{
304 \meth@checknotopen
305 \def\meth@postcond{\begin{list}{\textbf{\textpostcond}}{\meth@listdecl}
306 \item #1
307 \end{list}}
308 }

```

`\descr`

```

309 \newcommand{\descr}[1]{
310 \meth@checknotopen
311 \def\meth@descr{\begin{list}{\textbf{\textdescr}}{\meth@listdecl}
312 \item #1
313 \end{list}}
314 }

```

`\error`

```

315 \newcommand{\error}[1]{
316 \meth@checknotopen
317 \def\meth@error{\begin{list}{\textbf{\texterror}}{\meth@listdecl}
318 \item #1
319 \end{list}}
320 }

```

`\return`

```

321 \newcommand{\return}[1]{
322 \meth@checknotopen
323 \def\meth@return{\begin{list}{\textbf{\textreturn}}{\meth@listdecl}

```

```

324   \item #1
325   \end{list}}
326 }

\see
327 \newcommand{\see}[1]{
328   \meth@checknotopen
329   \def\meth@see{\begin{list}{\textbf{\textsee}}{\meth@listdecl}
330     \item #1
331     \end{list}}
332 }

\init
333 \newcommand{\init}[1]{
334   \meth@checknotopen
335   \def\meth@init{\begin{list}{\textbf{\textinit}}{\meth@listdecl}
336     \item #1
337     \end{list}}
338 }

\del
339 \newcommand{\del}[1]{
340   \meth@checknotopen
341   \def\meth@del{\begin{list}{\textbf{\textdel}}{\meth@listdecl}
342     \item #1
343     \end{list}}
344 }

```

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	<code>\headtabbed</code> <u>179</u>	<code>\meth@error</code> 114, 129, 317
<code>\'</code> 61, 63		<code>\meth@head</code> . 93, 120, 141, 149, 160, 170
D	<code>\init</code> <u>2</u> , <u>333</u>	<code>\meth@headparanum</code> . <u>31</u>
<code>data</code> (environment) <u>2</u> , <u>134</u>	M	<code>\meth@headtabbed</code> 94, 120, 181
<code>\DeclareOption</code> 42, 51, 59	<code>\meth@checkdoubleopen</code> <u>69</u> , 86, 136	<code>\meth@hpa</code> . 96, 191, 223
<code>\del</code> <u>2</u> , <u>339</u>	<code>\meth@checknotopen</code> <u>69</u> , 158, 180, 255, 292, 298, 304, 310, 316, 322, 328, 334, 340	<code>\meth@hpb</code> . 96, 192, 224
<code>\descr</code> <u>1</u> , <u>309</u>	<code>\meth@defheadpara</code> <u>221</u> , <u>254</u>	<code>\meth@hpc</code> . 96, 193, 225
E	<code>\meth@defpara</code> . <u>258</u> , 293	<code>\meth@hpd</code> . 96, 194, 226
environments:	<code>\meth@del</code> . 144, 153, 341	<code>\meth@hpe</code> . 97, 195, 227
<code>data</code> <u>2</u> , <u>134</u>	<code>\meth@descr</code> . . . 112, 127, 142, 151, 311	<code>\meth@hpf</code> . 97, 196, 228
<code>method</code> <u>1</u> , <u>84</u>		<code>\meth@hpg</code> . 97, 197, 229
<code>\error</code> <u>1</u> , <u>315</u>		<code>\meth@hph</code> . 97, 198, 230
<code>\ExecuteOptions</code> . . . 67		<code>\meth@hpi</code> . 98, 199, 231
H		<code>\meth@hpl</code> . 98, 200, 232
<code>\head</code> <u>1</u> , <u>157</u>		<code>\meth@hpk</code> . 98, 201, 233
<code>\headpara</code> 254		<code>\meth@hpl</code> . 98, 202, 234

<code>\meth@hpm</code>	. 99, 203, 235	<code>\meth@ph</code>	.. 105, 123, 267	<code>\meth@totwid</code>
<code>\meth@hpn</code>	. 99, 204, 236	<code>\meth@pi</code>	.. 106, 123, 268	33 , 88–90,
<code>\meth@hpo</code>	. 99, 205, 237	<code>\meth@pj</code>	.. 106, 123, 269	138, 139, 172, 184	
<code>\meth@hpp</code>	. 99, 206, 238	<code>\meth@pk</code>	.. 106, 123, 270	<code>\meth@where</code>
<code>\meth@hpq</code>	. 100, 207, 239	<code>\meth@pl</code>	.. 106, 123, 271	... 69 , 87, 132,	
<code>\meth@hpr</code>	. 100, 208, 240	<code>\meth@pm</code>	.. 107, 123, 272	137, 155, 159, 169	
<code>\meth@hps</code>	. 100, 209, 241	<code>\meth@pn</code>	.. 107, 123, 273	<code>method</code> (environment)	
<code>\meth@hpt</code>	. 100, 210, 242	<code>\meth@po</code>	.. 107, 124, 274	1 , 84
<code>\meth@hpu</code>	. 101, 211, 243	<code>\meth@postcond</code>		
<code>\meth@hpv</code>	. 101, 212, 244	113, 128, 305	P	
<code>\meth@hpw</code>	. 101, 213, 245	<code>\meth@pp</code>	.. 107, 124, 275	<code>\para</code> 1 , 291
<code>\meth@hpx</code>	. 101, 214, 246	<code>\meth@pq</code>	.. 108, 124, 276	<code>\postcond</code> 1 , 303
<code>\meth@hpy</code>	. 102, 215, 247	<code>\meth@pr</code>	.. 108, 124, 277	<code>\precond</code> 1 , 297
<code>\meth@hpz</code>	. 102, 216, 248	<code>\meth@precond</code>	<code>\ProcessOptions</code>	... 68
<code>\meth@indent</code>	111, 126, 299		
..	34 , 91, 162, 164	<code>\meth@ps</code>	.. 108, 124, 278	R	
<code>\meth@init</code>	143, 152, 335	<code>\meth@pt</code>	.. 108, 124, 279	<code>\relax</code> 68
<code>\meth@listdecl</code>	<code>\meth@pu</code>	.. 109, 124, 280	<code>\return</code> 1 , 321
..	34 , 293, 299,	<code>\meth@pv</code>	.. 109, 125, 281		
	305, 311, 317,	<code>\meth@pw</code>	.. 109, 125, 282	S	
	323, 329, 335, 341	<code>\meth@px</code>	.. 109, 125, 283	<code>\see</code> 1 , 327
<code>\meth@nameindent</code>	..	<code>\meth@py</code>	.. 110, 125, 284		
.....	39 , 186–188	<code>\meth@pz</code>	.. 110, 125, 285	T	
<code>\meth@namewid</code>	.. 39 ,	<code>\meth@return</code>	<code>\textdel</code> 42 , 341
	183, 185, 186, 190	115, 130, 323	<code>\textdescr</code> 42 , 311
<code>\meth@pa</code>	.. 104, 122, 260	<code>\meth@rightnamewid</code>	..	<code>\texterror</code> 42 , 317
<code>\meth@paranum</code> 31	..	39 , 184, 185, 189	<code>\textinit</code> 42 , 335
<code>\meth@pb</code>	.. 104, 122, 261	<code>\meth@righttotwid</code>	..	<code>\textpostcond</code>	.. 42 , 305
<code>\meth@pc</code>	.. 104, 122, 262	... 38 , 90, 91, 163		<code>\textprecond</code>	... 42 , 299
<code>\meth@pd</code>	.. 104, 122, 263	<code>\meth@see</code> 116,	<code>\textreturn</code> 42 , 323
<code>\meth@pe</code>	.. 105, 122, 264	131, 145, 154, 329		<code>\textsee</code> 42 , 329
<code>\meth@pf</code>	.. 105, 122, 265			V	
<code>\meth@pg</code>	.. 105, 122, 266			<code>\value</code> 222, 259

Change History

v1.0	General: First version by J. Wahlmann 1	v1.2	General: Change documentation to a dtx-file 1
v1.1	General: Facelifting by R. Gar- mann 1	v1.3	General: Documentation changes	.. 1
	init and del for the environment		data:	Change <code>\meth@totwid</code> from	
	data added 1		4 mm to 6 mm. 7
	New environment data for vari- ables 1	<code>\del:</code>	<code>\bf</code> replaced by <code>\textbf</code> .	.. 12
	Test for the use of the special commands outside the environ- ments 1	<code>\descr:</code>	<code>\bf</code> replaced by <code>\textbf</code> .	11
	method: Abzug von <code>\meth@totwid</code> von 4 mm auf 6 mm erhöht	.. 6	<code>\error:</code>	<code>\bf</code> replaced by <code>\textbf</code> .	11
			method:	Minimal change in the for- mat of typesetting 7
			<code>\head:</code>	<code>\tt</code> replaced by <code>\texttt</code> 8
			<code>\headtabbed:</code>	<code>\tt</code> replaced by	
			<code>\texttt</code>	8

<code>\init: \bf</code> replaced by <code>\textbf</code>	12	v1.5	
<code>\meth@checknotopen:</code> Show error messages with <code>\PackageError</code>	5		<code>method:</code> extend to 26 parameters 6, 7
<code>\meth@listdecl:</code> Minimale Änderung.	4	v1.6	<code>\meth@defheadpara:</code> extend to 26 parameters 9
<code>\para: \tt</code> replaced by <code>\texttt</code>	11		General: Documentation changes . 1
<code>\postcond: \bf</code> replaced by <code>\textbf</code>	11	v1.7	
<code>\precond: \bf</code> replaced by <code>\textbf</code>	11		General: Documentation changes . 1
<code>\return: \bf</code> replaced by <code>\textbf</code>	11	v1.8	First parts for internationalization 1
<code>\see: \bf</code> replaced by <code>\textbf</code>	12		General: Change standard to english 1
v1.4			
<code>\del:</code> insert missing braces.	12	v2.0	
<code>\descr:</code> insert missing braces.	11		General: First public version with internationalization and localization for english and german . 1
<code>\error:</code> insert missing braces.	11		
<code>\init:</code> insert missing braces.	12		
<code>\para:</code> insert braces, which were missing before	11	v2.0a	<code>\texterror:</code> French localization (thanks to Jean-Pierre Drucbert) 5
<code>\postcond:</code> insert missing braces.	11		
<code>\precond:</code> insert missing braces.	11		
<code>\return:</code> insert missing braces.	11	v2.0b	
<code>\see:</code> insert missing braces.	12		General: Change licence to lppl . . 1