

# The `setouterhbox` package

Heiko Oberdiek\*

2016/05/16 v1.8

## Abstract

If math stuff is set in an `\hbox`, then TeX performs some optimization and omits the implicate penalties `\binoppenalty` and `\relpenalty`. This packages tries to put stuff into an `\hbox` without getting lost of those penalties.

## Contents

<b>1</b>	<b>Documentation</b>	<b>2</b>
1.1	Introduction . . . . .	2
1.2	Acknowledgement . . . . .	2
1.3	Usage . . . . .	2
1.4	Option <code>hyperref</code> . . . . .	3
1.5	Example . . . . .	3
<b>2</b>	<b>Implementation</b>	<b>3</b>
2.1	Package start stuff . . . . .	3
2.2	Interface macros . . . . .	5
2.3	Main part . . . . .	6
2.4	Environment support . . . . .	8
2.5	Option <code>hyperref</code> . . . . .	9
<b>3</b>	<b>Installation</b>	<b>9</b>
3.1	Download . . . . .	9
3.2	Bundle installation . . . . .	9
3.3	Package installation . . . . .	9
3.4	Refresh file name databases . . . . .	10
3.5	Some details for the interested . . . . .	10
<b>4</b>	<b>References</b>	<b>10</b>
<b>5</b>	<b>History</b>	<b>11</b>
	[2005/10/05 v1.0] . . . . .	11
	[2005/10/07 v1.1] . . . . .	11
	[2005/10/18 v1.2] . . . . .	11
	[2006/02/12 v1.3] . . . . .	11
	[2006/08/26 v1.4] . . . . .	11
	[2007/04/26 v1.5] . . . . .	11
	[2007/05/17 v1.6] . . . . .	11
	[2007/09/09 v1.7] . . . . .	11
	[2016/05/16 v1.8] . . . . .	11
<b>6</b>	<b>Index</b>	<b>11</b>

---

\*Please report any issues at <https://github.com/ho-tex/oberdiek/issues>

# 1 Documentation

## 1.1 Introduction

There is a situation in `hyperref`'s driver for `dvips` where the user wants to have links that can be broken across lines. However `dvips` doesn't support the feature. With option `breaklinks` `hyperref` sets the links as usual, put them in a box and write the link data with box dimensions into the appropriate `\specials`. Then, however, it does not set the complete unbreakable box, but it unwrappes the material inside to allow line breaks. Of course line breaking and glue setting will falsify the link dimensions, but line breaking was more important for the user.

## 1.2 Acknowledgement

Jonathan Fine, Donald Arsenau and me discussed the problem in the newsgroup `comp.text.tex` where Damian Menscher has started the thread, see [1].

The discussion was productive and generated many ideas and code examples. In order to have a more permanent result I wrote this package and tried to implement most of the ideas, a kind of summary of the discussion. Thus I want and have to thank Jonathan Fine and Donald Arsenau very much.

Two weeks later David Kastrup (posting in `comp.text.tex`, [2]) remembered an old article of Michael Downes ([3]) in TUGboat, where Michael Downes already presented the method we discuss here. Nowadays we have  $\epsilon$ -TeX that extends the tool set of a TeX macro programmer. Especially useful  $\epsilon$ -TeX was in this package for detecting and dealing with erroneous situations.

However also nowadays a perfect solution for the problem is still missing at macro level. Probably someone has to go deep in the internals of the TeX compiler to implement a switch that let penalties stay where otherwise TeX would remove them for optimization reasons.

## 1.3 Usage

**Package loading.** L<sup>A</sup>TeX: as usually:

```
\usepackage{setouterhbox}
```

The package can also be included directly, thus plain TeX users write:

```
\input setouterhbox.sty
```

**Register allocation.** The material will be put into a box, thus we need to know these box number. If you need to allocate a new box register:

L<sup>A</sup>TeX: `\newsavebox{\langle name \rangle}`

plain TeX: `\newbox\langle name \rangle`

Then `\langle name \rangle` is a command that held the box number.

**Box wrapping.** L<sup>A</sup>TeX users put the material in the box with an environment similar to `lrbox`. The environment `setouterhbox` uses the same syntax and offers the same features, such as verbatim stuff inside:

```
\begin{setouterhbox}{\langle box number \rangle}... \end{setouterhbox}
```

Users with plain TeX do not have environments, they use instead:

```
\setouterhbox{\langle box number \rangle}... \endsetouterhbox
```

In both cases the material is put into an `\hbox` and assigned to the given box, denoted by `\langle box number \rangle`. Note the assignment is local, the same way `lrbox` behaves.

**Unwrapping.** The box material is ready for unwrapping:

```
\unhbox{box number}
```

## 1.4 Option hyperref

Package `url` uses math mode for typesetting urls. Break points are inserted by `\binoppenalty` and `\relpenalty`. Unhappily these break points are removed, if `hyperref` is used with option `breaklinks` and drivers that depend on `pdfmark:dvips`, `vtexpdfmark`, `textures`, and `dvipson`. Thus the option `hyperref` enables the method of this package to avoid the removal of `\relpenalty` and `\binoppenalty`. Thus you get more break points. However, the link areas are still wrong for these drivers, because they are not supporting broken links.

Note, you need version 2006/08/16 v6.75c of package `hyperref`, because starting with this version the necessary hook is provided that package `setouterhbox` uses.

```
\usepackage[...]{hyperref}[2006/08/16]
\usepackage[hyperref]{setouterhbox}
```

Package order does not matter.

## 1.5 Example

```
1 (*example)
2 \documentclass[a5paper]{article}
3 \usepackage[url][2005/06/27]
4 \usepackage{setouterhbox}
5
6 \newsavebox{\testbox}
7
8 \setlength{\parindent}{0pt}
9 \setlength{\parskip}{2em}
10
11 \begin{document}
12 \raggedright
13
14 \url{http://this.is.a.very.long.host.name/followed/%
15 by/a/very_long_long_long_path.html}%
16
17 \sbox\testbox{%
18   \url{http://this.is.a.very.long.host.name/followed/%
19   by/a/very_long_long_long_path.html}%
20 }%
21 \unhbox\testbox
22
23 \begin{setouterhbox}{\testbox}%
24   \url{http://this.is.a.very.long.host.name/followed/%
25   by/a/very_long_long_long_path.html}%
26 \end{setouterhbox}
27 \unhbox\testbox
28
29 \end{document}
30 </example>
```

## 2 Implementation

Internal macros are prefixed by `\setouterhbox`, `@` is not used inside names, thus we do not need to care of its catcode if we are not using it as L<sup>A</sup>T<sub>E</sub>X package.

### 2.1 Package start stuff

```
31 (*package)
```

Prevent reloading more than one, necessary for plain T<sub>E</sub>X: Reload check, especially if the package is not used with L<sup>A</sup>T<sub>E</sub>X.

```

32 \begingroup\catcode61\catcode48\catcode32=10\relax%
33 \catcode13=5 % ^M
34 \endlinechar=13 %
35 \catcode35=6 % #
36 \catcode39=12 % '
37 \catcode44=12 % ,
38 \catcode45=12 % -
39 \catcode46=12 % .
40 \catcode58=12 % :
41 \catcode64=11 % @
42 \catcode123=1 % {
43 \catcode125=2 % }
44 \expandafter\let\expandafter\x\csname ver@setouterhbox.sty\endcsname
45 \ifx\x\relax % plain-TeX, first loading
46 \else
47 \def\empty{}%
48 \ifx\x\empty % LaTeX, first loading,
49 % variable is initialized, but \ProvidesPackage not yet seen
50 \else
51 \expandafter\ifx\csname PackageInfo\endcsname\relax
52 \def\x#1#2{%
53 \immediate\write-1{Package #1 Info: #2.}%
54 }%
55 \else
56 \def\x#1#2{\PackageInfo{#1}{#2, stopped}}%
57 \fi
58 \x{setouterhbox}{The package is already loaded}%
59 \aftergroup\endinput
60 \fi
61 \fi
62 \endgroup%

```

Package identification:

```

63 \begingroup\catcode61\catcode48\catcode32=10\relax%
64 \catcode13=5 % ^M
65 \endlinechar=13 %
66 \catcode35=6 % #
67 \catcode39=12 % '
68 \catcode40=12 % (
69 \catcode41=12 % )
70 \catcode44=12 % ,
71 \catcode45=12 % -
72 \catcode46=12 % .
73 \catcode47=12 % /
74 \catcode58=12 % :
75 \catcode64=11 % @
76 \catcode91=12 % [
77 \catcode93=12 % ]
78 \catcode123=1 % {
79 \catcode125=2 % }
80 \expandafter\ifx\csname ProvidesPackage\endcsname\relax
81 \def\x#1#2#3[#4]{\endgroup
82 \immediate\write-1{Package: #3 #4}%
83 \xdef#1{#4}%
84 }%
85 \else
86 \def\x#1#2[#3]{\endgroup
87 #2[#3]}%
88 \ifx#1\undefined
89 \xdef#1{#3}%
90 \fi

```

```

91     \ifx#1\relax
92     \xdef#1{#3}%
93     \fi
94   }%
95 \fi
96 \expandafter\x\csname ver@setouterhbox.sty\endcsname
97 \ProvidesPackage{setouterhbox}%
98 [2016/05/16 v1.8 Set hbox in outer horizontal mode (H0)]%
99 \begingroup\catcode61\catcode48\catcode32=10\relax%
100 \catcode13=5 % ^~M
101 \endlinechar=13 %
102 \catcode123=1 % {
103 \catcode125=2 % }
104 \catcode64=11 % @
105 \def\x{\endgroup
106   \expandafter\edef\csname setouterhboxAtEnd\endcsname{%
107     \endlinechar=\the\endlinechar\relax
108     \catcode13=\the\catcode13\relax
109     \catcode32=\the\catcode32\relax
110     \catcode35=\the\catcode35\relax
111     \catcode61=\the\catcode61\relax
112     \catcode64=\the\catcode64\relax
113     \catcode123=\the\catcode123\relax
114     \catcode125=\the\catcode125\relax
115   }%
116 }%
117 \x\catcode61\catcode48\catcode32=10\relax%
118 \catcode13=5 % ^~M
119 \endlinechar=13 %
120 \catcode35=6 % #
121 \catcode64=11 % @
122 \catcode123=1 % {
123 \catcode125=2 % }
124 \def\TMP@EnsureCode#1#2{%
125   \edef\setouterhboxAtEnd{%
126     \setouterhboxAtEnd
127     \catcode#1=\the\catcode#1\relax
128   }%
129   \catcode#1=#2\relax
130 }
131 \TMP@EnsureCode{40}{12}% (
132 \TMP@EnsureCode{41}{12}% )
133 \TMP@EnsureCode{44}{12}% ,
134 \TMP@EnsureCode{45}{12}% -
135 \TMP@EnsureCode{46}{12}% .
136 \TMP@EnsureCode{47}{12}% /
137 \TMP@EnsureCode{58}{12}% :
138 \TMP@EnsureCode{60}{12}% <
139 \TMP@EnsureCode{62}{12}% >
140 \TMP@EnsureCode{91}{12}% [
141 \TMP@EnsureCode{93}{12}% ]
142 \TMP@EnsureCode{96}{12}% ‘
143 \edef\setouterhboxAtEnd{\setouterhboxAtEnd\noexpand\endinput}

```

## 2.2 Interface macros

`\setouterhboxBox` The method requires a global box assignment. To be on the safe side, a new box register is allocated for this global box assignment.

```
144 \newbox\setouterhboxBox
```

`\setouterhboxFailure` Error message for both plain  $\TeX$  and  $\LaTeX$

```
145 \begingroup\expandafter\expandafter\expandafter\endgroup
146 \expandafter\ifx\csname RequirePackage\endcsname\relax
```

```

147 \input infwarerr.sty\relax
148 \else
149 \RequirePackage{infwarerr}[2016/05/16]%
150 \fi
151 \edef\setouterhboxFailure#1#2{%
152 \expandafter\noexpand\csname @PackageError\endcsname
153 {setouterhbox}{#1}{#2}%
154 }

```

## 2.3 Main part

eTeX provides much better means for checking error conditions. Thus lines marked by "E" are executed if eTeX is available, otherwise the lines marked by "T" are used.

```

155 \begingroup\expandafter\expandafter\expandafter\endgroup
156 \expandafter\ifx\csname lastnodetype\endcsname\relax
157 \catcode'T=9 % ignore
158 \catcode'E=14 % comment
159 \else
160 \catcode'T=14 % comment
161 \catcode'E=9 % ignore
162 \fi

```

`\setouterhboxRemove` Remove all kern, glue, and penalty nodes; poor man's version, if  $\varepsilon$ -TeX is not available

```

163 \def\setouterhboxRemove{%
164 E \ifnum\lastnodetype<11 %
165 E \else
166 E \ifnum\lastnodetype>13 %
167 E \else
168 \unskip\unkern\unpenalty
169 E \expandafter\expandafter\expandafter\setouterhboxRemove
170 E \fi
171 E \fi
172 }%

```

`\setouterhbox` Passing the box contents by macro parameter would prevent catcode changes in the box contents like by `\verb`. Also `\bgroup` and `\egroup` does not work, because stuff has to be added at the begin and end of the box, thus the syntax `\setouterhbox{<box number>}\endsetouterhbox` is used. Also we automatically get an environment `setouterhbox` if L<sup>A</sup>T<sub>E</sub>X is used.

```

173 \def\setouterhbox#1{%
174 \begingroup
175 \def\setouterhboxNum{#1}%
176 \setbox0\vbox\bgroup
177 T \kern.123pt\relax % marker
178 T \kern0pt\relax % removed by \setouterhboxRemove
179 \begingroup
180 \everypar{}%
181 \noindent
182 }

```

`\endsetouterhbox` Most of the work is done in the end part, thus the heart of the method follows:

```

183 \def\endsetouterhbox{%
184 \endgroup

```

Omit the first pass to get the penalties of the second pass.

```

185 \pretolerance-1 %

```

We don't want a third pass with `\emergencystretch`.

```

186 \tolerance10000 %
187 \hsizemaxdimen

```

Line is not underfull:

```
188      \parfillskip Opt plus 1filll\relax
189      \leftskipOpt\relax
```

Suppress underfull \hbox warnings, is explicit line breaks are used.

```
190      \rightskipOpt plus 1fil\relax
191      \everypar{}%
```

Ensure that there is a paragraph and prevents \endgraph from eating terminal glue:

```
192      \kernOpt%
193      \endgraf
194      \setouterhboxRemove
195 E      \ifnum\lastnodetype=1 %
196 E          \global\setbox\setouterhboxBox\lastbox
197 E          \loop
198 E              \setouterhboxRemove
199 E          \ifnum\lastnodetype=1 %
200 E              \setbox0=\lastbox
201 E              \global\setbox\setouterhboxBox=\hbox{%
202 E                  \unhbox0 %
```

Remove \rightskip, a penalty with -10000 is part of the previous line.

```
203 E          \unskip
204 E          \unhbox\setouterhboxBox
205 E      }%
206 E      \repeat
207 E      \else
208 E          \setouterhboxFailure{%
209 E              Something is wrong%
210 E          }{%
211 E              Could not find expected line.%
212 E              \MessageBreak
213 E              (\string\lastnodetype: \number\lastnodetype, expected: 1)%
214 E          }%
215 E      \fi
216 E      \setouterhboxRemove
217 T      \global\setbox\setouterhboxBox\lastbox
218 T      \loop
219 T          \setouterhboxRemove
220 T          \setbox0=\lastbox
221 T      \ifcase\ifvoid0 1\else0\fi
222 T          \global\setbox\setouterhboxBox=\hbox{%
223 T              \unhbox0 %
```

Remove \rightskip, a penalty with -10000 is part of the previous line.

```
224 T          \unskip
225 T          \unhbox\setouterhboxBox
226 T      }%
227 T      \repeat
228 T      \ifdim.123pt=\lastkern
229 T      \else
230 T          \setouterhboxFailure{%
231 T              Something is wrong%
232 T          }{%
233 T              Unexpected stuff was detected before the line.%
234 T          }%
235 T      \fi
236 T      \egroup
237 T      \ifcase \ifnum\wd0=0 \else 1\fi
238 T          \ifdim\ht0=.123pt \else 1\fi
239 T          \ifnum\dp0=0 \else 1\fi
240 T          0 %
241 E      \ifnum\lastnodetype=-1 %
```

There was just one line that we have caught.

```
242     \else
243     \setouterhboxFailure{%
244     Something is wrong%
245     }{%
246     After fetching the line there is more unexpected stuff.%
247 E     \MessageBreak
248 E     (\string\lastnodetype: \number\lastnodetype, expected: -1)%
249     }%
250     \fi
251 E \egroup
252 \expandafter\endgroup
253 \expandafter\setouterhboxFinish\expandafter{%
254 \number\setouterhboxNum
255 }%
256 }
```

## 2.4 Environment support

Check `\@currenvir` for the case that `\setouterhbox` was called as environment. Then the box assignment must be put after the `\endgroup` of `\end{...}`.

```
257 \def\setouterhboxCurr{setouterhbox}
258 \def\setouterhboxLast#1{%
259 \setbox#1\hbox{%
260 \unhbox\setouterhboxBox
261 \unskip % remove \rightskip glue
262 \unskip % remove \parfillskip glue
263 \unpenalty % remove paragraph ending \penalty 10000
264 \unkern % remove explicit kern inserted above
265 }%
266 }
```

`\setouterhboxFinish #1` is an explicit number.

```
267 \def\setouterhboxFinish#1{%
268 \begingroup\expandafter\expandafter\expandafter\endgroup
269 \expandafter\ifx\csname @currenvir\endcsname\setouterhboxCurr
270 \aftergroup\setouterhboxLast
271 \aftergroup{%
272 \setouterhboxAfter #1\NIL
273 \aftergroup}%
274 \else
275 \setouterhboxLast{#1}%
276 \fi
277 }
```

`\setouterhboxAfter #1` is an explicit number.

```
278 \def\setouterhboxAfter#1#2\NIL{%
279 \aftergroup#1%
280 \ifx\#2\%
281 \else
282 \setouterhboxReturnAfterFi{%
283 \setouterhboxAfter#2\NIL
284 }%
285 \fi
286 }
```

`\setouterhboxReturnAfterFi` A utility macro to get tail recursion.

```
287 \long\def\setouterhboxReturnAfterFi#1\fi{\fi#1}
```

Restore catcodes we have need to distinguish between the implementation with and without  $\varepsilon$ -TeX.

```
288 \catcode69=11\relax % E
289 \catcode84=11\relax % T
```



## 2.5 Option hyperref

```
290 \begingroup
291   \def\x{LaTeX2e}%
292 \expandafter\endgroup
293 \ifx\x\fmtname
294 \else
295   \expandafter\setouterhboxAtEnd
296 \fi%
```

`\Hy@setouterhbox` `\Hy@setouterhbox` is the internal hook that hyperref uses since 2006/02/12 v6.75a.

```
297 \DeclareOption{hyperref}{%
298   \long\def\Hy@setouterhbox#1#2{%
299     \setouterhbox{#1}#2\endsetouterhbox
300   }%
301 }

302 \ProcessOptions\relax
303 \setouterhboxAtEnd%
304 </package>
```

## 3 Installation

### 3.1 Download

**Package.** This package is available on CTAN<sup>1</sup>:

[CTAN:macros/latex/contrib/oberdiek/setouterhbox.dtx](#) The source file.

[CTAN:macros/latex/contrib/oberdiek/setouterhbox.pdf](#) Documentation.

**Bundle.** All the packages of the bundle ‘oberdiek’ are also available in a TDS compliant ZIP archive. There the packages are already unpacked and the documentation files are generated. The files and directories obey the TDS standard.

[CTAN:install/macros/latex/contrib/oberdiek.tds.zip](#)

TDS refers to the standard “A Directory Structure for T<sub>E</sub>X Files” ([CTAN:pkg/tds](#)). Directories with `texmf` in their name are usually organized this way.

### 3.2 Bundle installation

**Unpacking.** Unpack the `oberdiek.tds.zip` in the TDS tree (also known as `texmf` tree) of your choice. Example (linux):

```
unzip oberdiek.tds.zip -d ~/texmf
```

### 3.3 Package installation

**Unpacking.** The `.dtx` file is a self-extracting docstrip archive. The files are extracted by running the `.dtx` through plain T<sub>E</sub>X:

```
tex setouterhbox.dtx
```

---

<sup>1</sup>[CTAN:pkg/setouterhbox](#)

**TDS.** Now the different files must be moved into the different directories in your installation TDS tree (also known as `texmf` tree):

```
setouterhbox.sty      → tex/generic/oberdiek/setouterhbox.sty
setouterhbox.pdf      → doc/latex/oberdiek/setouterhbox.pdf
setouterhbox-example.tex → doc/latex/oberdiek/setouterhbox-example.tex
setouterhbox.dtx      → source/latex/oberdiek/setouterhbox.dtx
```

If you have a `docstrip.cfg` that configures and enables `docstrip`'s TDS installing feature, then some files can already be in the right place, see the documentation of `docstrip`.

### 3.4 Refresh file name databases

If your  $\TeX$  distribution ( $\TeX$  Live, MiK $\TeX$ , ...) relies on file name databases, you must refresh these. For example,  $\TeX$  Live users run `texhash` or `mktexlsr`.

### 3.5 Some details for the interested

**Unpacking with  $\LaTeX$ .** The `.dtx` chooses its action depending on the format:

**plain  $\TeX$ :** Run `docstrip` and extract the files.

**$\LaTeX$ :** Generate the documentation.

If you insist on using  $\LaTeX$  for `docstrip` (really, `docstrip` does not need  $\LaTeX$ ), then inform the autodetect routine about your intention:

```
latex \let\install=y\input{setouterhbox.dtx}
```

Do not forget to quote the argument according to the demands of your shell.

**Generating the documentation.** You can use both the `.dtx` or the `.drv` to generate the documentation. The process can be configured by the configuration file `ltxdoc.cfg`. For instance, put this line into this file, if you want to have A4 as paper format:

```
\PassOptionsToClass{a4paper}{article}
```

An example follows how to generate the documentation with `pdf $\LaTeX$` :

```
pdflatex setouterhbox.dtx
makeindex -s gind.ist setouterhbox.idx
pdflatex setouterhbox.dtx
makeindex -s gind.ist setouterhbox.idx
pdflatex setouterhbox.dtx
```

## 4 References

- [1] Damian Menscher, [news:comp.text.tex](https://groups.google.com/group/comp.text.tex/msg/79648d4cf1f8bc13), *overlong lines in List of Figures*, <dh058t\$`qbd`\$1@news.ks.uiuc.edu>, 23rd September 2005. <https://groups.google.com/group/comp.text.tex/msg/79648d4cf1f8bc13>
- [2] David Kastrup, [news:comp.text.tex](https://groups.google.com/group/comp.text.tex/msg/7cf0a345ef932e52), *Re: ANN: outerhbox.sty – collect horizontal material, for unboxing into a paragraph*, <85y8551rx3.fsf@lola.goethe.zz>, 7th October 2005. <https://groups.google.com/group/comp.text.tex/msg/7cf0a345ef932e52>
- [3] Michael Downes, *Line breaking in `\unboxed Text`*, TUGboat 11 (1990), pp. 605–612.
- [4] Sebastian Rahtz, Heiko Oberdiek: *The hyperref package*; 2006/08/16 v6.75c; [CTAN:pkg/hyperref](https://ctan.org/pkg/hyperref).

## 5 History

[2005/10/05 v1.0]

- First version.

[2005/10/07 v1.1]

- Option hyperref added.

[2005/10/18 v1.2]

- Support for explicit line breaks added.

[2006/02/12 v1.3]

- DTX format.
- Documentation extended.

[2006/08/26 v1.4]

- Date of hyperref updated.

[2007/04/26 v1.5]

- Use of package infwarerr.

[2007/05/17 v1.6]

- Standard header part for generic files.

[2007/09/09 v1.7]

- Catcode section added.

[2016/05/16 v1.8]

- Documentation updates.

## 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; plain numbers refer to the code lines where the entry is used.

	<b>Symbols</b>	<i>72, 73, 74, 75, 76, 77, 78, 79,</i>
<code>\@undefined</code> .....	<i>88</i>	<i>99, 100, 102, 103, 104, 108, 109,</i>
<code>\</code> .....	<i>280</i>	<i>110, 111, 112, 113, 114, 117,</i>
	<b>A</b>	<i>118, 120, 121, 122, 123, 127,</i>
<code>\aftergroup</code> ....	<i>59, 270, 271, 273, 279</i>	<i>129, 157, 158, 160, 161, 288, 289</i>
	<b>B</b>	<code>\csname</code> .....
<code>\begin</code> .....	<i>11, 23</i>	<i>44,</i>
	<b>C</b>	<i>51, 80, 96, 106, 146, 152, 156, 269</i>
<code>\catcode</code> .....	<i>32, 33, 35,</i>	<b>D</b>
	<i>36, 37, 38, 39, 40, 41, 42, 43,</i>	<code>\DeclareOption</code> .....
	<i>63, 64, 66, 67, 68, 69, 70, 71,</i>	<i>297</i>
		<code>\documentclass</code> .....
		<i>2</i>
		<code>\dp</code> .....
		<i>239</i>

<b>E</b>		<code>\pretolerance</code> . . . . . 185	
<code>\empty</code> . . . . .	47, 48	<code>\ProcessOptions</code> . . . . . 302	
<code>\end</code> . . . . .	26, 29	<code>\ProvidesPackage</code> . . . . . 49, 97	
<code>\endcsname</code> . . . . .	44,	<b>R</b>	
	51, 80, 96, 106, 146, 152, 156, 269	<code>\raggedright</code> . . . . . 12	
<code>\endgraf</code> . . . . .	193	<code>\repeat</code> . . . . . 206, 227	
<code>\endinput</code> . . . . .	59, 143	<code>\RequirePackage</code> . . . . . 149	
<code>\endlinechar</code> . . . . .	34, 65, 101, 107, 119	<code>\rightskip</code> . . . . . 190, 261	
<code>\endsetouterhbox</code> . . . . .	183, 299	<b>S</b>	
<code>\everypar</code> . . . . .	180, 191	<code>\sbox</code> . . . . . 17	
<b>F</b>		<code>\setbox</code> . . . . . 176,	
<code>\fmtname</code> . . . . .	293	196, 200, 201, 217, 220, 222, 259	
<b>H</b>		<code>\setlength</code> . . . . . 8, 9	
<code>\hbox</code> . . . . .	201, 222, 259	<code>\setouterhbox</code> . . . . . 173, 299	
<code>\hsize</code> . . . . .	187	<code>\setouterhboxAfter</code> . . . . . 272, 278	
<code>\ht</code> . . . . .	238	<code>\setouterhboxAtEnd</code> . . . . .	
<code>\Hy@setouterhbox</code> . . . . .	297	125, 126, 143, 295, 303	
<b>I</b>		<code>\setouterhboxBox</code> . . . . . 144,	
<code>\ifcase</code> . . . . .	221, 237	196, 201, 204, 217, 222, 225, 260	
<code>\ifdim</code> . . . . .	228, 238	<code>\setouterhboxCurr</code> . . . . . 257, 269	
<code>\ifnum</code> 164, 166, 195, 199, 237, 239, 241		<code>\setouterhboxFailure</code> 145, 208, 230, 243	
<code>\ifvoid</code> . . . . .	221	<code>\setouterhboxFinish</code> . . . . . 253, 267	
<code>\ifx</code> . . . . .	45, 48, 51,	<code>\setouterhboxLast</code> . . . . . 258, 270, 275	
	80, 88, 91, 146, 156, 269, 280, 293	<code>\setouterhboxNum</code> . . . . . 175, 254	
<code>\immediate</code> . . . . .	53, 82	<code>\setouterhboxRemove</code> . . . . .	
<code>\input</code> . . . . .	147	163, 178, 194, 198, 216, 219	
<b>K</b>		<code>\setouterhboxReturnAfterFi</code> . 282, 287	
<code>\kern</code> . . . . .	177, 178, 192	<b>T</b>	
<b>L</b>		<code>\testbox</code> . . . . . 6, 17, 21, 23, 27	
<code>\lastbox</code> . . . . .	196, 200, 217, 220	<code>\the</code> . . . . . 107, 108,	
<code>\lastkern</code> . . . . .	228	109, 110, 111, 112, 113, 114, 127	
<code>\lastnodetype</code> . . . . .		<code>\TMP@EnsureCode</code> . . . . .	
	164, 166, 195, 199, 213, 241, 248	124, 131, 132, 133, 134, 135,	
<code>\leftskip</code> . . . . .	189	136, 137, 138, 139, 140, 141, 142	
<code>\loop</code> . . . . .	197, 218	<code>\tolerance</code> . . . . . 186	
<b>M</b>		<b>U</b>	
<code>\maxdimen</code> . . . . .	187	<code>\unhbox</code> . 21, 27, 202, 204, 223, 225, 260	
<code>\MessageBreak</code> . . . . .	212, 247	<code>\unkern</code> . . . . . 168, 264	
<b>N</b>		<code>\unpenalty</code> . . . . . 168, 263	
<code>\newbox</code> . . . . .	144	<code>\unskip</code> . . . . . 168, 203, 224, 261, 262	
<code>\newsavebox</code> . . . . .	6	<code>\url</code> . . . . . 14, 18, 24	
<code>\NIL</code> . . . . .	272, 278, 283	<code>\usepackage</code> . . . . . 3, 4	
<code>\noindent</code> . . . . .	181	<b>V</b>	
<code>\number</code> . . . . .	213, 248, 254	<code>\vbox</code> . . . . . 176	
<b>P</b>		<b>W</b>	
<code>\PackageInfo</code> . . . . .	56	<code>\wd</code> . . . . . 237	
<code>\parfillskip</code> . . . . .	188, 262	<code>\write</code> . . . . . 53, 82	
<code>\parindent</code> . . . . .	8	<b>X</b>	
<code>\parskip</code> . . . . .	9	<code>\x</code> . . . . . 44, 45, 48, 52,	
<code>\penalty</code> . . . . .	263	56, 58, 81, 86, 96, 105, 117, 291, 293	