

# The `setouterhbox` package

Heiko Oberdiek  
<oberdiek@uni-freiburg.de>

2007/09/09 v1.7

## 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 . . . . .	5
2.4	Environment support . . . . .	7
2.5	Option <code>hyperref</code> . . . . .	8
<b>3</b>	<b>Test</b>	<b>8</b>
3.1	Catcode checks for loading . . . . .	8
3.2	Test with package <code>url</code> . . . . .	9
<b>4</b>	<b>Installation</b>	<b>10</b>
4.1	Download . . . . .	10
4.2	Bundle installation . . . . .	10
4.3	Package installation . . . . .	10
4.4	Refresh file name databases . . . . .	11
4.5	Some details for the interested . . . . .	11
<b>5</b>	<b>References</b>	<b>11</b>
<b>6</b>	<b>History</b>	<b>11</b>
	[2005/10/05 v1.0] . . . . .	11
	[2005/10/07 v1.1] . . . . .	12
	[2005/10/18 v1.2] . . . . .	12
	[2006/02/12 v1.3] . . . . .	12
	[2006/08/26 v1.4] . . . . .	12
	[2007/04/26 v1.5] . . . . .	12
	[2007/05/17 v1.6] . . . . .	12
	[2007/09/09 v1.7] . . . . .	12

# 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  $\varepsilon$ -TeX that extends the tool set of a TeX macro programmer. Especially useful  $\varepsilon$ -TeX was in this package for detecting and dealing with errorneous 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>T<sub>E</sub>X: 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>T<sub>E</sub>X:** `\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>T<sub>E</sub>X 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 `dvipsone`. 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- $\text{\TeX}$ : Reload check, especially if the package is not used with  $\text{\LaTeX}$ .

```

32 \begingroup
33   \catcode44 12 % ,
34   \catcode45 12 % -
35   \catcode46 12 % .
36   \catcode58 12 % :
37   \catcode64 11 % @
38   \expandafter\let\expandafter\x\csname ver@setouterhbox.sty\endcsname
39   \ifcase 0%
40     \ifx\x\relax % plain
41     \else
42       \ifx\x\empty % LaTeX
43       \else
44         1%
45       \fi
46     \fi
47   \else
48     \expandafter\ifx\csname PackageInfo\endcsname\relax
49     \def\x#1#2{%
50       \immediate\write-1{Package #1 Info: #2.}%
51     }%
52   \else
53     \def\x#1#2{\PackageInfo{#1}{#2, stopped}}%
54   \fi
55   \x{setouterhbox}{The package is already loaded}%
56 \endgroup
57 \expandafter\endinput
58 \fi
59 \endgroup

```

Package identification:

```

60 \begingroup
61   \catcode40 12 % (
62   \catcode41 12 % )
63   \catcode44 12 % ,
64   \catcode45 12 % -
65   \catcode46 12 % .
66   \catcode47 12 % /
67   \catcode58 12 % :
68   \catcode64 11 % @
69   \expandafter\ifx\csname ProvidesPackage\endcsname\relax
70     \def\x#1#2#3[#4]{\endgroup
71       \immediate\write-1{Package: #3 #4}%
72       \xdef#1{#4}%
73     }%
74   \else
75     \def\x#1#2[#3]{\endgroup
76       #2[#{#3}]%
77       \ifx#1\relax
78         \xdef#1{#3}%
79       \fi
80     }%
81   \fi
82 \expandafter\x\csname ver@setouterhbox.sty\endcsname
83 \ProvidesPackage{setouterhbox}%
84 [2007/09/09 v1.7 Set hbox in outer horizontal mode (H0)]
85 \expandafter\edef\csname setouterhboxAtEnd\endcsname{%
86   \catcode64 \the\catcode64\relax
87 }
88 \catcode64 11 % @
89 \def\TMP@EnsureCode#1#2{%
90   \edef\setouterhboxAtEnd{%

```

```

91     \setouterhboxAtEnd
92     \catcode#1 \the\catcode#1\relax
93 }%
94 \catcode#1 #2\relax
95 }
96 \TMP@EnsureCode{40}{12}% (
97 \TMP@EnsureCode{41}{12}% )
98 \TMP@EnsureCode{44}{12}% ,
99 \TMP@EnsureCode{45}{12}% -
100 \TMP@EnsureCode{46}{12}% .
101 \TMP@EnsureCode{47}{12}% /
102 \TMP@EnsureCode{58}{12}% :
103 \TMP@EnsureCode{60}{12}% <
104 \TMP@EnsureCode{61}{12}% =
105 \TMP@EnsureCode{62}{12}% >
106 \TMP@EnsureCode{96}{12}% ‘

```

## 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.

```
107 \newbox\setouterhboxBox
```

`\setouterhboxFailure` Error message for both plain- $\text{\TeX}$  and  $\text{\LaTeX}$

```

108 \begingroup\expandafter\expandafter\expandafter\endgroup
109 \expandafter\ifx\csname RequirePackage\endcsname\relax
110   \input infwarerr.sty\relax
111 \else
112   \RequirePackage{infwarerr}[2007/09/09]%
113 \fi
114 \edef\setouterhboxFailure#1#2{%
115   \expandafter\noexpand\csname @PackageError\endcsname
116     {setouterhbox}{#1}{#2}%
117 }

```

## 2.3 Main part

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

```

118 \begingroup\expandafter\expandafter\expandafter\endgroup
119 \expandafter\ifx\csname lastnodetype\endcsname\relax
120   \catcode'T=9 % ignore
121   \catcode'E=14 % comment
122 \else
123   \catcode'T=14 % comment
124   \catcode'E=9 % ignore
125 \fi

```

`\setouterhboxRemove` Remove all kern, glue, and penalty nodes; poor man's version, if  $\text{\eTeX}$  is not available

```

126 \def\setouterhboxRemove{%
127 E \ifnum\lastnodetype<11 %
128 E   \else
129 E   \ifnum\lastnodetype>13 %
130 E   \else
131     \unskip\unkern\unpenalty
132 E   \expandafter\expandafter\expandafter\setouterhboxRemove
133 E   \fi
134 E \fi
135 }%

```

`\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  $\text{\LaTeX}$  is used.

```

136 \def\setouterhbox#1{%
137   \begingroup
138   \def\setouterhboxNum{#1}%
139   \setbox0\vbox\bgroup
140 T   \kern.123pt\relax % marker
141 T   \kern0pt\relax % removed by \setouterhboxRemove
142   \begingroup
143   \everypar{}%
144   \noindent
145 }
```

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

```

146 \def\endsetouterhbox{%
147   \endgroup
```

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

```
148   \pretolerance-1 %
```

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

```

149   \tolerance10000 %
150   \hsize\maxdimen
```

Line is not underfull:

```

151   \parfillskip 0pt plus 1filll\relax
152   \leftskip0pt\relax
```

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

```

153   \rightskip0pt plus 1fil\relax
154   \everypar{}%
```

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

```

155   \kern0pt%
156   \endgraf
157   \setouterhboxRemove
158 E   \ifnum\lastnodetype=1 %
159 E     \global\setbox\setouterhboxBox\lastbox
160 E     \loop
161 E       \setouterhboxRemove
162 E       \ifnum\lastnodetype=1 %
163 E         \setbox0=\lastbox
164 E         \global\setbox\setouterhboxBox=\hbox{%
165 E           \unhbox0 %
```

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

```

166 E       \unskip
167 E       \unhbox\setouterhboxBox
168 E     }%
169 E   \repeat
170 E \else
171 E   \setouterhboxFailure{%
172 E     Something is wrong%
173 E   }{%
174 E     Could not find expected line.%
175 E     \MessageBreak
176 E     (\string\lastnodetype: \number\lastnodetype, expected: 1)%
177 E   }%
178 E \fi
179 E \setouterhboxRemove
```

```

180 T    \global\setbox\setouterhboxBox\lastbox
181 T    \loop
182 T        \setouterhboxRemove
183 T        \setbox0=\lastbox
184 T    \ifcase\ifvoid0 1\else0\fi
185 T        \global\setbox\setouterhboxBox=\hbox{%
186 T            \unhbox0 %

```

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

```

187 T        \unskip
188 T        \unhbox\setouterhboxBox
189 T    }%
190 T    \repeat
191 T    \ifdim.123pt=\lastkern
192 T    \else
193 T        \setouterhboxFailure{%
194 T            Something is wrong%
195 T        }{%
196 T            Unexpected stuff was detected before the line.%
197 T        }%
198 T    \fi
199 T    \egroup
200 T    \ifcase \ifnum\wd0=0 \else 1\fi
201 T        \ifdim\ht0=.123pt \else 1\fi
202 T        \ifnum\dp0=0 \else 1\fi
203 T        0 %
204 E    \ifnum\lastnodetype=-1 %

```

There was just one line that we have caught.

```

205        \else
206            \setouterhboxFailure{%
207                Something is wrong%
208            }{%
209                After fetching the line there is more unexpected stuff.%
210 E            \MessageBreak
211 E            (\string\lastnodetype: \number\lastnodetype, expected: -1)%
212        }%
213    \fi
214 E    \egroup
215    \expandafter\endgroup
216    \expandafter\setouterhboxFinish\expandafter{%
217        \number\setouterhboxNum
218    }%
219 }

```

## 2.4 Environment support

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

```

220 \def\setouterhboxCurr{\setouterhbox}
221 \def\setouterhboxLast#1{%
222     \setbox#1\hbox{%
223         \unhbox\setouterhboxBox
224         \unskip % remove \rightskip glue
225         \unskip % remove \parfillskip glue
226         \unpenalty % remove paragraph ending \penalty 10000
227         \unkern % remove explicit kern inserted above
228     }%
229 }

```

\setouterhboxFinish #1 is an explicit number.

```

230 \def\setouterhboxFinish#1{%
231     \begingroup\expandafter\expandafter\expandafter\endgroup

```

```

232 \expandafter\ifx\csname @currentenv\endcsname\setouterhboxCurr
233 \aftergroup\setouterhboxLast
234 \aftergroup{%
235 \setouterhboxAfter #1\NIL
236 \aftergroup}%
237 \else
238 \setouterhboxLast{#1}%
239 \fi
240 }

```

`\setouterhboxAfter` #1 is an explicit number.

```

241 \def\setouterhboxAfter#1#2\NIL{%
242 \aftergroup#1%
243 \ifx\#2\%
244 \else
245 \setouterhboxReturnAfterFi{%
246 \setouterhboxAfter#2\NIL
247 }%
248 \fi
249 }

```

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

```

250 \long\def\setouterhboxReturnAfterFi#1\fi{\fi#1}

```

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

```

251 \catcode69=11\relax % E
252 \catcode84=11\relax % T

```

## 2.5 Option hyperref

```

253 \begingroup
254 \def\x{LaTeX2e}%
255 \expandafter\endgroup
256 \ifx\x\fmtname
257 \else
258 \setouterhboxAtEnd
259 \expandafter\endinput
260 \fi

```

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

```

261 \DeclareOption{hyperref}{%
262 \long\def\Hy@setouterhbox#1#2{%
263 \setouterhbox{#1}#2\endsetouterhbox
264 }%
265 }

```

```

266 \ProcessOptions\relax
267 \setouterhboxAtEnd
268 \endpackage

```

## 3 Test

### 3.1 Catcode checks for loading

```

269 (*test1)
270 \catcode'\@=11 %
271 \def\RestoreCatcodes{}
272 \count@=0 %
273 \loop

```

```

274 \edef\RestoreCatcodes{%
275   \RestoreCatcodes
276   \catcode\the\count@=\the\catcode\count@\relax
277 }%
278 \ifnum\count@<255 %
279   \advance\count@\@ne
280 \repeat
281
282 \def\RangeCatcodeInvalid#1#2{%
283   \count@=#1\relax
284   \loop
285     \catcode\count@=15 %
286     \ifnum\count@<#2\relax
287       \advance\count@\@ne
288     \repeat
289 }
290 \def\Test{%
291   \RangeCatcodeInvalid{0}{47}%
292   \RangeCatcodeInvalid{58}{64}%
293   \RangeCatcodeInvalid{91}{96}%
294   \RangeCatcodeInvalid{123}{255}%
295   \catcode'\@=12 %
296   \catcode'\=0 %
297   \catcode'\{=1 %
298   \catcode'\}=2 %
299   \catcode'\#=6 %
300   \catcode'\[=12 %
301   \catcode'\]=12 %
302   \catcode'\%=14 %
303   \catcode'\ =10 %
304   \catcode13=5 %
305   \input setouterhbox.sty\relax
306   \RestoreCatcodes
307 }
308 \Test
309 \csname @@end\endcsname
310 \end
311 </test1>

```

### 3.2 Test with package url

```

312 <*test2>
313 \nofiles
314 \documentclass[a5paper]{article}
315 \usepackage[url]{2005/06/27}
316 \usepackage{setouterhbox}
317
318 \newsavebox{\testbox}
319
320 \setlength{\parindent}{0pt}
321 \setlength{\parskip}{2em}
322
323 \begin{document}
324 \raggedright
325
326 \url{http://this.is.a.very.long.host.name/followed/%
327 by/a/very_long_long_long_path.html}%
328
329 \sbox\testbox{%
330   \url{http://this.is.a.very.long.host.name/followed/%
331   by/a/very_long_long_long_path.html}%
332 }%
333 \unhbox\testbox

```

```

334
335 \begin{setouterhbox}{\testbox}%
336   \url{http://this.is.a.very.long.host.name/followed/%
337   by/a/very_long_long_long_path.html}%
338 \end{setouterhbox}
339 \unhbox\testbox
340
341 \end{document}
342 </test2>

```

## 4 Installation

### 4.1 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:macros/latex/contrib/oberdiek/oberdiek-tds.zip](#)

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

### 4.2 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
```

**Script installation.** Check the directory `TDS:scripts/oberdiek/` for scripts that need further installation steps. Package `attachfile2` comes with the Perl script `pdfatfi.pl` that should be installed in such a way that it can be called as `pdfatfi`. Example (linux):

```

chmod +x scripts/oberdiek/pdfatfi.pl
cp scripts/oberdiek/pdfatfi.pl /usr/local/bin/

```

### 4.3 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
```

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

<code>setouterhbox.sty</code>	→ <code>tex/generic/oberdiek/setouterhbox.sty</code>
<code>setouterhbox.pdf</code>	→ <code>doc/latex/oberdiek/setouterhbox.pdf</code>
<code>setouterhbox-example.tex</code>	→ <code>doc/latex/oberdiek/setouterhbox-example.tex</code>
<code>setouterhbox-test1.tex</code>	→ <code>doc/latex/oberdiek/setouterhbox-test1.tex</code>
<code>setouterhbox-test2.tex</code>	→ <code>doc/latex/oberdiek/setouterhbox-test2.tex</code>
<code>setouterhbox.dtx</code>	→ <code>source/latex/oberdiek/setouterhbox.dtx</code>

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`.

<sup>1</sup><http://ftp.ctan.org/tex-archive/>

## 4.4 Refresh file name databases

If your  $\text{T}_{\text{E}}\text{X}$  distribution (te $\text{T}_{\text{E}}\text{X}$ , mik $\text{T}_{\text{E}}\text{X}$ , ...) relies on file name databases, you must refresh these. For example, te $\text{T}_{\text{E}}\text{X}$  users run `texhash` or `mktextlsr`.

## 4.5 Some details for the interested

**Attached source.** The PDF documentation on CTAN also includes the `.dtx` source file. It can be extracted by AcrobatReader 6 or higher. Another option is `pdftk`, e.g. unpack the file into the current directory:

```
pdftk setouterhbox.pdf unpack_files output .
```

**Unpacking with  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ .** The `.dtx` chooses its action depending on the format:

**plain- $\text{T}_{\text{E}}\text{X}$ :** Run `docstrip` and extract the files.

**$\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ :** Generate the documentation.

If you insist on using  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  for `docstrip` (really, `docstrip` does not need  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ ), 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 $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$ :

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

## 5 References

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

## 6 History

[2005/10/05 v1.0]

- First version.

- Option hyperref added.

- Support for explicit line breaks added.

- DTX format.
- Documentation extended.

- Date of hyperref updated.

- Use of package `infwarerr`.

- Standard header part for generic files.

- Catcode section added.

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.

12

<b>H</b>		
\hbox	164, 185, 222	\repeat 169, 190, 280, 288
\hsize	150	\RequirePackage 112
\ht	201	\RestoreCatcodes 271, 274, 275, 306
\Hy@setouterhbox	261	\rightskip 153, 224
<b>I</b>		<b>S</b>
\ifcase	39, 184, 200	\sbox 17, 329
\ifdim	191, 201	\setbox 139,
\ifnum	127, 129,	159, 163, 164, 180, 183, 185, 222
158, 162, 200, 202, 204, 278, 286		\setlength 8, 9, 320, 321
\ifvoid	184	\setouterhbox 136, 263
\ifx	40, 42,	\setouterhboxAfter 235, 241
48, 69, 77, 109, 119, 232, 243, 256		\setouterhboxAtEnd 90, 91, 258, 267
\immediate	50, 71	\setouterhboxBox 107,
\input	110, 305	159, 164, 167, 180, 185, 188, 223
<b>K</b>		\setouterhboxCurr 220, 232
\kern	140, 141, 155	\setouterhboxFailure 108, 171, 193, 206
<b>L</b>		\setouterhboxFinish 216, 230
\lastbox	159, 163, 180, 183	\setouterhboxLast 221, 233, 238
\lastkern	191	\setouterhboxNum 138, 217
\lastnodetype	127, 129, 158, 162, 176, 204, 211	\setouterhboxRemove 126, 141, 157, 161, 179, 182
\leftskip	152	\setouterhboxReturnAfterFi 245, 250
\loop	160, 181, 273, 284	
<b>M</b>		<b>T</b>
\maxdimen	150	\Test 290, 308
\MessageBreak	175, 210	\testbox 6, 17,
<b>N</b>		21, 23, 27, 318, 329, 333, 335, 339
\newbox	107	\the 86, 92, 276
\newsavebox	6, 318	\TMP@EnsureCode 89, 96, 97, 98,
\NIL	235, 241, 246	99, 100, 101, 102, 103, 104, 105, 106
\nofiles	313	\tolerance 149
\noindent	144	
\number	176, 211, 217	<b>U</b>
<b>P</b>		\unhbox 21,
\PackageInfo	53	27, 165, 167, 186, 188, 223, 333, 339
\parfillskip	151, 225	\unkern 131, 227
\parindent	8, 320	\unpenalty 131, 226
\parskip	9, 321	\unskip 131, 166, 187, 224, 225
\penalty	226	\url 14, 18, 24, 326, 330, 336
\pretolerance	148	\usepackage 3, 4, 315, 316
\ProcessOptions	266	
\ProvidesPackage	83	<b>V</b>
<b>R</b>		\vbox 139
\raggedright	12, 324	
\RangeCatcodeInvalid	282, 291, 292, 293, 294	<b>W</b>
		\wd 200
		\write 50, 71
		<b>X</b>
		\x 38, 40,
		42, 49, 53, 55, 70, 75, 82, 254, 256