



logbox*

ϵ -TeX showbox facilities for exploration purpose

FC

2011/03/26 – version 1.0

Abstract

Exploring a box content and its dimensions is often required for typography is made of measurements. `logbox` provides a few macros to achieve this task easily: `\viewbox*` is the most useful.

`logbox` is intended for exploration purpose, this is not a package for typesetting.

ϵ -TeX is obviously required !

Contents of logbox

1	Five exploration commands	1	v1.0 [2011/03/26]
2	IMPLEMENTATION	4	
3	History	4	
4	Index	5	

List of listings and examples

Example of <code>\logbox</code>	2	<code>\viewbox*</code> gives all the informations .	2
Example of <code>\dimbox</code>	2		

1 Five exploration commands

`\logbox` \langle box register \rangle | \langle number \rangle

`\logbox` displays the contents of a TeX box register in the `.log` file.

This is exactly `\showbox` with:

- `\showoutput` set if `\tracingoutput = 0`
If `\tracingoutput > 0` it is assumed that the user set the tracing level he desired before `\logbox`.
- The minimum value for `\showboxbreadth` and `\showboxdepth` is assumed to be 10
- The ϵ -TeX `\interactionmode` is temporarily set to 2 (`\scrollmode`) and restored to its previous value just after the execution of `\showbox`: `\logbox` does not stop the compilation, just reports the box contents into the `.log` file.

* This documentation is produced with the DocStrip utility.
 → To get the package, run: `etex logbox.dtx`
 → To get the documentation run (thrice): `pdflatex logbox.dtx`
 To get the index, run: `makeindex -s gind.ist logbox.idx`
 The `.dtx` file is embedded into this `.pdf` file thank to `embedfile` by H. Oberdiek.

Example 1: Example of `\logbox`

```
\setbox 0 =\hbox{ Hello }\logbox 0
```

```
logbox: 0 on input line 55
> \box0=
\hbox(7.54149+0.0)x31.75774
.\glue 3.63054 plus 1.81337 minus 1.20892
.\T1/cmr/m/n/10.95 H
.\T1/cmr/m/n/10.95 e
.\T1/cmr/m/n/10.95 l
.\T1/cmr/m/n/10.95 l
.\T1/cmr/m/n/10.95 o
.\glue 3.63054 plus 1.81337 minus 1.20892

! OK.
```

`\dimbox` *<box register>* | *<number>*

`\dimbox` typeset the type (`\hbox`, `\vbox` or void box) and dimensions of a box in the document itself:

Example 2: Example of `\dimbox`

```
\setbox 0 = \hbox { Hello }\dimbox 0 \par
\setbox 1 = \vtop {\hsize=60mm Hello \par (World)}\dimbox 1
```

```
hbox0: wd=31.75774pt ht=7.54149pt dp=0.0pt tot=7.54149pt
vbox1: wd=170.71652pt ht=7.54149pt dp=16.3369pt tot=23.87839pt
```

`\viewbox` *<box register>* | *<number>* `\viewbox*` *<box register>* | *<number>*

`\viewbox` basically does `\dimbox` and `\logbox`.

`\viewbox*` prints a copy of the box in the document, after the dimensions have been reported:

Example 3: `\viewbox*` gives all the informations

```
\setbox 0 =\vtop {\hsize=60mm Hello \par (World)}
\viewbox*0
```

```
vbox0: wd=170.71652pt ht=7.54149pt dp=16.3369pt tot=23.87839pt
Hello
(World)
```

And the .log file contains (artificially splitted here on two columns):

```
logbox: 0 on input line 1
> \box0=
\vbox(7.54149+16.3369)x170.71652
.\hbox(7.54149+0.0)x170.71652, glue set 146.21986fil.\kern-0.9067
..\hbox(0.0+0.0)x0.0
..\rule(0.0+*)x0.0
..\T1/cmr/m/n/10.95 H
..\T1/cmr/m/n/10.95 e
..\T1/cmr/m/n/10.95 l
..\T1/cmr/m/n/10.95 l
..\T1/cmr/m/n/10.95 o
..\penalty 10000
..\glue(\parfillskip) 0.0 plus 1.0fil
..\glue(\rightskip) 0.0
.\glue(\parskip) 0.0
.\glue(\baselineskip) 5.38942
.\hbox(8.21059+2.7369)x170.71652, glue set 133.2164fil
..\hbox(0.0+0.0)x0.0
..\T1/cmr/m/n/10.95 (
..\T1/cmr/m/n/10.95 W
..\kern-0.9067
..\T1/cmr/m/n/10.95 o
..\T1/cmr/m/n/10.95 r
..\T1/cmr/m/n/10.95 l
..\T1/cmr/m/n/10.95 d
..\T1/cmr/m/n/10.95 )
..\penalty 10000
..\glue(\parfillskip) 0.0 plus 1.0fil
..\glue(\rightskip) 0.0
! OK.
```

`\ShowGroups`

The command `\ShowGroups` executes the primitive `\showgroups` but the compilation does not stop.

\ShowLists

The command `\ShowLists` executes the primitive `\showlists` but the compilation does not stop.

\ShowIfs

The command `\ShowIfs` executes the primitive `\showifs` but the compilation does not stop.

2 IMPLEMENTATION

Identification

The package namespace is `\logb@x`

```

1 \*package
2 \NeedsTeXFormat{LaTeX2e}[2005/12/01]
3 \ProvidesPackage{logbox}
4     [2011/03/26 v1.0 -  $\varepsilon$ -TeX showbox facilities for exploration (FC)]

```

\logbox \TeX `\showbox` without stop in the compilation.

```

5 \protected\def\logbox {\begingroup \afterassignment\logb@x@log \count@ }
6 \def\logb@x@log {\ifnum\tracingoutput>0 \ifnum\showboxdepth<2 \showboxdepth=2 \fi
7     \ifnum\showboxbreadth<10 \showboxbreadth=10 \fi
8     \else \showoutput \fi
9     \message {^^Jlogbox: \the\count@ \oneline^^J}\expandafter \scrollmode
10    \expandafter \showbox \expandafter \count@ \expandafter
11    \interactionmode\the\interactionmode \endgroup
12 }% \logb@x@log

```

\dimbox Typeset the type and dimensions of a box in the document.

```

13 \protected\def\dimbox {\begingroup \afterassignment\logb@x@dim \count@ }
14 \def\logb@x@dim {\nfss@text
15     {\ifhbox\count@ hbox\else\ifvbox\count@ vbox\else box\fi\fi
16     \the\count@: \ifvoid\count@ void\else wd=\the\wd\count@\
17     ht=\the\ht\count@\
18     dp=\the\dp\count@\
19     tot=\the\dimexpr\ht\count@ +\dp\count@ \relax\fi } \endgroup
20 }% \logb@x@dim

```

\viewbox Typeset the type and dimensions of a box in the document and inserts the content of that box.

```

21 \protected\def\viewbox {\begingroup \@ifstar
22     {\def\logb@x@ {\@@par\copy\count@ }\afterassignment\logb@x@view \count@ }
23     {\let\logb@x@ \relax \afterassignment\logb@x@view \count@ }}
24 \def\logb@x@view {\dimbox\count@ \logb@x@ \logb@x@log }

```

\Viewbox Same as `\viewbox` but the content of the box is given as argument, rather than the box register.

```

25 \protected\def\Viewbox {\@ifstar \logb@x@View \logb@x@View }
26 \long\def\logb@x@View #1{\begingroup \setbox0=\hbox {#1}\viewbox*0\endgroup }

```

ShowGroups ε -TeX `\showgroups` without stop in the compilation.

```

27 \def>ShowGroups {\expandafter \scrollmode \expandafter \showgroups \expandafter
28     \interactionmode\the\interactionmode \relax }

```

\ShowLists ε -TeX `\showlists` without stop in the compilation.

```

29 \def>ShowLists {\expandafter \scrollmode \expandafter \showlists \expandafter
30     \interactionmode\the\interactionmode \relax }

```

\ShowIfs ε -TeX `\showgroups` without stop in the compilation.

```

31 \def>ShowIfs {\expandafter \scrollmode \expandafter \showifs \expandafter
32     \interactionmode\the\interactionmode \relax }
33 \*package

```

3 History

[2011/03/26 v1.0]

- First version. But it works well yet !

4 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.

Symbols	I	S
<code>\@@par</code> 22	<code>\ifhbox</code> 15	<code>\scrollmode</code> ... 9, 27, 29, 31
<code>\@ifstar</code> 21, 25	<code>\ifnum</code> 6, 7	<code>\setbox</code> 26
	<code>\ifvbox</code> 15	<code>\showbox</code> 10
	<code>\ifvoid</code> 16	<code>\showboxbreadth</code> 7
<code>\□</code> 16, 17, 18	<code>\interactionmode</code>	<code>\showboxdepth</code> 6
 11, 28, 30, 32	<code>\ShowGroups</code> 2, <u>27</u> , 27
A	L	T
<code>\afterassignment</code> 5, 13, 22, 23	<code>\logb@x@</code> 22, 23, 24	<code>\showgroups</code> 27
	<code>\logb@x@dim</code> 13, 14, 20	<code>\ShowIfs</code> 3, <u>31</u>
C	<code>\logb@x@log</code> ... 5, 6, 12, 24	<code>\showifs</code> 31
<code>\copy</code> 22	<code>\logb@x@View</code> 25, 26	<code>\ShowLists</code> 3, <u>29</u>
<code>\count@</code> .. 5, 9, 10, 13, 15, 16, 17, 18, 19, 22, 23, 24	<code>\logb@x@view</code> ... 22, 23, 24	<code>\showlists</code> 29
	<code>\logbox</code> 1, <u>5</u>	<code>\showoutput</code> 8
D	M	V
<code>\dimbox</code> 2, <u>13</u> , 24	<code>\message</code> 9	<code>\Viewbox</code> <u>25</u>
<code>\dimexpr</code> 19		<code>\viewbox</code> 2, <u>21</u> , 26
<code>\dp</code> 18, 19	N	W
	<code>\nfss@text</code> 14	
E	O	
<code>\expandafter</code> 9, 10, 27, 29, 31	<code>\on@line</code> 9	
H	P	
<code>\hbox</code> 26	<code>\protected</code> 5, 13, 21, 25	
<code>\ht</code> 17, 19		<code>\wd</code> 16