

# The auxhook package

Heiko Oberdiek  
<oberdiek@uni-freiburg.de>

2006/05/31 v1.0

## Abstract

Package `auxhook` provides hooks for adding stuff at the begin of `.aux` files.

## Contents

<b>1</b>	<b>User interface</b>	<b>1</b>
<b>2</b>	<b>Implementation</b>	<b>2</b>
2.1	Identification . . . . .	2
2.2	Hook setup . . . . .	2
2.3	User macros . . . . .	2
2.4	Patches . . . . .	3
2.4.1	<code>\document</code> . . . . .	3
2.4.2	<code>\@include</code> . . . . .	3
<b>3</b>	<b>Installation</b>	<b>4</b>
3.1	Some details for the interested . . . . .	4
<b>4</b>	<b>References</b>	<b>5</b>
<b>5</b>	<b>History</b>	<b>5</b>
	[2006/05/31 v1.0] . . . . .	5
<b>6</b>	<b>Index</b>	<b>5</b>

## 1 User interface

There are two kinds of `.aux` files, the main `.aux` file and the `.aux` file that belongs to an included file, specified by `\include`.

Some packages write macros in the auxiliary files. If the user stops using the package, these macros will usually cause error messages because of unknown commands. Prominent example is package `babel`'s `\select@language`.

But such a package could be written more cooperative. It can also provide a definition in the auxiliary file (`\providecommand`) that silently disables the macros of the package if the package is no longer in use.

In case of the main auxiliary file, `\AtBeginDocument` can be used for this purpose. Especially if several packages are involved, the order cannot be controlled always (e.g., see package `hypdestopt` that hooks into `hyperref`'s macros). And there isn't any hook for the auxiliary files of the `\include` feature.

Thus this package patches L<sup>A</sup>T<sub>E</sub>X's macros `\document` and `\@include` to add the hooks where the auxiliary files are opened and the first line with `\relax` is written.

The patching can fail, if these macros are redefined by some other package. If the other package still uses the original definition, then load package `auxhook`

earlier. (With `\RequirePackage` the package also can be loaded before the class). If the redefinition doesn't use the original meaning, then you can try to load package `auxhook` afterwards, but you need luck that the patch succeeds.

The hooks are macros:

`\@beginmainauxhook`: Start of the main auxiliary file. The hook is called after the first line with `\relax` is written.

`\@beginpartauxhook`: The same for the auxiliary files that belongs to the files that are included by `\include`.

If you want to add something to these hooks, you can use `\g@addto@macro` from L<sup>A</sup>T<sub>E</sub>X's kernel. But the package provides macros to add code that adds a line to the auxiliary file:

```
\AddLineBeginMainAux {line}
\AddLineBeginPartAux {line}
\AddLineBeginAux {line}
```

The *line* is added at the begin of the main auxiliary file by `\AddLineBeginMainAux` and at the begin of the auxiliary files of included files by `\AddLineBeginPartAux`. `\AddLineBeginAux` writes in both kinds of auxiliary files.

Examples, see packages `hypdestopt` ([1]) and `zref` ([3]).

## 2 Implementation

### 2.1 Identification

```
1 <*package>
2 \NeedsTeXFormat{LaTeX2e}
3 \ProvidesPackage{auxhook}%
4 [2006/05/31 v1.0 Hooks for auxiliary files (H0)]
```

### 2.2 Hook setup

`\@beginmainauxhook` The hook for the main auxiliary file, initially empty.

```
5 \providecommand*{\@beginmainauxhook}{}%
```

`\@beginpartauxhook` The hook for auxiliary files of included files, initially empty.

```
6 \providecommand*{\@beginpartauxhook}{}%
```

### 2.3 User macros

`\AddLineBeginMainAux`

```
7 \newcommand{\AddLineBeginMainAux}[1]{%
8   \g@addto@macro\@beginmainauxhook{%
9     \immediate\write\@mainaux{#1}%
10  }%
11 }
```

`\AtBeginPartAuxLine`

```
12 \newcommand{\AddLineBeginPartAux}[1]{%
13   \g@addto@macro\@beginpartauxhook{%
14     \immediate\write\@partaux{#1}%
15   }%
16 }
```

`\AddLineBeginAux`

```

17 \newcommand{\AddLineBeginAux}[1]{%
18   \AddLineBeginMainAux{#1}%
19   \AddLineBeginPartAux{#1}%
20 }

```

## 2.4 Patches

### 2.4.1 `\document`

```

21 \begingroup
22   \def\y#1\immediate\write\@mainaux#2#3\auxhook@nil{%
23     \toks@{%
24       #1\immediate\write\@mainaux{#2}%
25       \@beginmainauxhook
26       #3%
27     }%
28     \xdef\document{\the\toks@}%
29   \endgroup
30 }%
31 \def\x#1\immediate\write\@mainaux#2#3\auxhook@nil{%
32   \toks@{#3}%
33   \edef\x{\the\toks@}%
34   \ifx\x\@empty
35     \PackageWarningNoLine{auxhook}{%
36       Cannot patch \string\document,\MessageBreak
37       use \string\AtBeginDocument\space instead%
38     }%
39   \endgroup
40   \AtBeginDocument{%
41     \if@filesw
42       \@beginmainauxhook
43     \fi
44   }%
45   \else
46     \expandafter\y\document\auxhook@nil
47   \fi
48 }%
49 \expandafter\x\document\immediate\write\@mainaux{}\auxhook@nil

```

### 2.4.2 `\@include`

```

50 \begingroup
51   \def\y#1\immediate\write\@partaux#2#3\auxhook@nil{%
52     \endgroup
53     \def\@include##1 {%
54       #1\immediate\write\@partaux{#2}%
55       \@beginpartauxhook
56       #3%
57     }%
58   }%
59   \def\x#1\immediate\write\@partaux#2#3\auxhook@nil{%
60     \toks@{#3}%
61     \edef\x{\the\toks@}%
62     \ifx\x\@empty
63       \PackageWarningNoLine{auxhook}{%
64         Cannot patch \string\@include,\MessageBreak
65         patch dropped%
66       }%
67     \endgroup
68   \else
69     \expandafter\y\@include{##1} \auxhook@nil
70   \fi

```

```

71 }%
72 \expandafter\x\@include{#1} \immediate\write\@partaux{}\auxhook@nil
73 \</package>

```

## 3 Installation

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

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

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

**Unpacking.** The `.dtx` file is a self-extracting `docstrip` archive. The files are extracted by running the `.dtx` through plain-`TEX`:

```
tex auxhook.dtx
```

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

```

auxhook.sty → tex/latex/oberdiek/auxhook.sty
auxhook.pdf → doc/latex/oberdiek/auxhook.pdf
auxhook.dtx → source/latex/oberdiek/auxhook.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`.

**Refresh file databases.** If your `TEX` distribution (`teTEX`, `mikTEX`, ...) rely on file databases, you must refresh these. For example, `teTEX` users run `texhash` or `mktexlsr`.

### 3.1 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 auxhook.pdf unpack_files output .
```

**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{auxhook.dtx}
```

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

---

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

**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 pdfL<sup>A</sup>T<sub>E</sub>X:

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

## 4 References

- [1] Heiko Oberdiek: *The hypdestopt package*; 2006/05/30 v1.0; [CTAN:macros/latex/contrib/oberdiek/hypdestopt.pdf](#).
- [2] Sebastian Rahtz, Heiko Oberdiek: *The hyperref package*; 2006/08/16 v6.75c; [CTAN:macros/latex/contrib/hyperref/](#).
- [3] Heiko Oberdiek: *The zref package*; 2006/05/25 v1.2; [CTAN:macros/latex/contrib/oberdiek/zref.pdf](#).

## 5 History

[2006/05/31 v1.0]

- First version.

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

Symbols		I	
<code>\@beginmainauxhook</code>	5, 8, 25, 42	<code>\if@files</code>	41
<code>\@beginpartauxhook</code>	6, 13, 55	<code>\ifx</code>	34, 62
<code>\@empty</code>	34, 62	<code>\immediate</code>	9,
<code>\@include</code>	53, 64, 69, 72		14, 22, 24, 31, 49, 51, 54, 59, 72
<code>\@mainaux</code>	9, 22, 24, 31, 49	M	
<code>\@partaux</code>	14, 51, 54, 59, 72	<code>\MessageBreak</code>	36, 64
A		N	
<code>\AddLineBeginAux</code>	17	<code>\NeedsTeXFormat</code>	2
<code>\AddLineBeginMainAux</code>	2, 7, 18	<code>\newcommand</code>	7, 12, 17
<code>\AddLineBeginPartAux</code>	12, 19	P	
<code>\AtBeginDocument</code>	37, 40	<code>\PackageWarningNoLine</code>	35, 63
<code>\AtBeginPartAuxLine</code>	12	<code>\providecommand</code>	5, 6
<code>\auxhook@nil</code>	22, 31, 46, 49, 51, 59, 69, 72	<code>\ProvidesPackage</code>	3
D		S	
<code>\document</code>	28, 36, 46, 49	<code>\space</code>	37
G		T	
<code>\g@addto@macro</code>	8, 13	<code>\the</code>	28, 33, 61
		<code>\toks@</code>	23, 28, 32, 33, 60, 61

	W		Y
\write	9, 14, 22, 24, 31, 49, 51, 54, 59, 72	\y	22, 46, 51, 69
	X		
\x	31, 33, 34, 49, 59, 61, 62, 72		