

The auxhook package

Heiko Oberdiek
<heiko.oberdiek at gmail.com>

2009/12/14 v1.2

Abstract

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

Contents

1	User interface	1
2	Implementation	2
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
3	Installation	4
3.1	Download	4
3.2	Bundle installation	4
3.3	Package installation	5
3.4	Refresh file name databases	5
3.5	Some details for the interested	5
4	References	6
5	History	6
	[2006/05/31 v1.0]	6
	[2007/04/06 v1.1]	6
	[2009/12/14 v1.2]	6
6	Index	6

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 \LaTeX '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 \LaTeX '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 [2009/12/14 v1.2 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   \@ifundefined{beamer@origdocument}{%
23     \def\auxhook@document{\document}%
24   }{%
25     \def\auxhook@document{\beamer@origdocument}%
26   }%
27   \long\def\y#1\immediate\write\@mainaux#2#3\auxhook@nil{%
28     \toks@{%
29       #1\immediate\write\@mainaux{#2}%
30       \@beginmainauxhook
31       #3%
32     }%
33     \expandafter\xdef\auxhook@document{\the\toks@}%
34     \endgroup
35   }%
36   \long\def\x#1\immediate\write\@mainaux#2#3\auxhook@nil{%
37     \toks@{#3}%
38     \edef\x{\the\toks@}%
39     \ifx\x@empty
40       \PackageWarningNoLine{auxhook}{%
41         Cannot patch \expandafter\string\auxhook@document,%
42         \MessageBreak
43         using \string\AtBeginDocument\space instead%
44       }%
45       \endgroup
46       \AtBeginDocument{%
47         \if@files
48           \@beginmainauxhook
49         \fi
50       }%
51     \else
52       \expandafter\expandafter\expandafter\y\auxhook@document
53       \auxhook@nil
54     \fi
55   }%
56   \expandafter\expandafter\expandafter\x\auxhook@document
57   \immediate\write\@mainaux{}\auxhook@nil
```

2.4.2 \@include

```

58 \begingroup
59   \long\def\y#1\immediate\write\@partaux#2#3\auxhook@nil#4{%
60     \endgroup
61     \def#4##1 {%
62       #1\immediate\write\@partaux{#2}%
63       \@beginpartauxhook
```

```

64     #3%
65 }%
66 }%
67 \long\def\x#1\immediate\write\@partaux#2#3\auxhook@nil#4{%
68   \toks@{#3}%
69   \edef\x{\the\toks@}%
70   \ifx\x\@empty
71     \PackageWarningNoLine{auxhook}{%
72       Cannot patch \string#4,\MessageBreak
73       patch dropped%
74     }%
75   \endgroup
76   \else
77     \expandafter\y#4{##1} \auxhook@nil#4%
78   \fi
79 }%
80 \@ifundefined{ReFiCh@org@include}{%
81   \expandafter\x\@include{#1} %
82   \immediate\write\@partaux{}\auxhook@nil\@include
83 }{%
84   \expandafter\x\ReFiCh@org@include{#1} %
85   \immediate\write\@partaux{}\auxhook@nil\ReFiCh@org@include
86 }%
87 \endpackage

```

3 Installation

3.1 Download

Package. This package is available on CTAN¹:

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

[CTAN:macros/latex/contrib/oberdiek/auxhook.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 \TeX Files” ([CTAN:tds/tds.pdf](#)). 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
```

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/
```

¹<http://ftp.ctan.org/tex-archive/>

3.3 Package installation

Unpacking. The `.dtx` file is a self-extracting `docstrip` archive. The files are extracted by running the `.dtx` through plain $\mathrm{T}_{\mathrm{E}}\mathrm{X}$:

```
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`.

3.4 Refresh file name databases

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

3.5 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 auxhook.pdf unpack_files output .
```

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

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

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

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

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 $\mathrm{L}^{\mathrm{A}}\mathrm{T}_{\mathrm{E}}\mathrm{X}$` :

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

[2007/04/06 v1.1]

- Fix for class beamer.

[2009/12/14 v1.2]

- Support for package rerunfilecheck added (`\@include`).

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.

Symbols	M
<code>\@beginmainauxhook</code> 5 , 8 , 30 , 48	<code>\MessageBreak</code> 42 , 72
<code>\@beginpartauxhook</code> 6 , 13 , 63	
<code>\@empty</code> 39 , 70	N
<code>\@ifundefined</code> 22 , 80	<code>\NeedsTeXFormat</code> 2
<code>\@include</code> 81 , 82	<code>\newcommand</code> 7 , 12 , 17
<code>\@mainaux</code> 9 , 27 , 29 , 36 , 57	P
<code>\@partaux</code> 14 , 59 , 62 , 67 , 82 , 85	<code>\PackageWarningNoLine</code> 40 , 71
A	<code>\providecommand</code> 5 , 6
<code>\AddLineBeginAux</code> 17	<code>\ProvidesPackage</code> 3
<code>\AddLineBeginMainAux</code> 2 , 7 , 18	R
<code>\AddLineBeginPartAux</code> 12 , 19	<code>\ReFiCh@org@include</code> 84 , 85
<code>\AtBeginDocument</code> 43 , 46	S
<code>\AtBeginPartAuxLine</code> 12	<code>\space</code> 43
<code>\auxhook@document</code> 23 , 25 , 33 , 41 , 52 , 56	T
<code>\auxhook@nil</code> 27 , 36 , 53 , 57 , 59 , 67 , 77 , 82 , 85	<code>\the</code> 33 , 38 , 69
B	<code>\toks@</code> 28 , 33 , 37 , 38 , 68 , 69
<code>\beamer@origdocument</code> 25	W
D	<code>\write</code> 9 , 14 , 27 , 29 , 36 , 57 , 59 , 62 , 67 , 82 , 85
<code>\document</code> 23	X
G	<code>\x</code> 36 , 38 , 39 , 56 , 67 , 69 , 70 , 81 , 84
<code>\g@addto@macro</code> 8 , 13	Y
I	<code>\y</code> 27 , 52 , 59 , 77
<code>\if@files</code> 47	
<code>\ifx</code> 39 , 70	
<code>\immediate</code> 9 , 14 , 27 , 29 , 36 , 57 , 59 , 62 , 67 , 82 , 85	