

# The classlist package

Heiko Oberdiek  
<heiko.oberdiek at gmail.com>

2008/08/11 v1.3

## Abstract

This package records the loaded classes and stores them in a list.

## Contents

<b>1 Documentation</b>	<b>1</b>
1.1 Background	1
1.2 Usage	1
<b>2 Implementation</b>	<b>2</b>
<b>3 Installation</b>	<b>4</b>
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
<b>4 History</b>	<b>5</b>
[2005/06/19 v1.0]	5
[2005/06/19 v1.1]	6
[2006/02/20 v1.2]	6
[2008/08/11 v1.3]	6
<b>5 Index</b>	<b>6</b>

## 1 Documentation

### 1.1 Background

This packages is an answer of a newsgroup question:

```
Newsgroup: comp.text.tex
Subject:   Finding the Document Class
From:     Herber Schulz
Date:     18 Jun 2005 13:16:49 -0500
Message-ID: <herbs-D55DB9.13170418062005@news.isp.giganews.com>
```

### 1.2 Usage

Load this package before \documentclass:

```
\RequirePackage{classlist}
\documentclass[some,options]{whatever}
```

It then records the classes with options.

If used after `\documentclass`, `\@filelist` is parsed for classes. The additional data specified options and requested version is no longer available here.

`\MainClass` contains the first loaded class.

`\ClassList` stores the class entries, eg.

```
\ClassList → \ClassListEntry{myarticle}{a4paper}{}
              \ClassListEntry{article}{}{}
```

`\ClassListEntry` has three arguments:

```
#1:  class name
#2:  options given in \documentclass/\LoadClass
#3:  requested version, not the version of class
```

`\PrintClassList` prints the list on screen it can be configured by

`\PrintClassListTitle` for the title and

`\PrintClassListEntry` for formatting the entries. See the implementation how to use these.

## 2 Implementation

```
1 (*package)
Package identification.
2 \NeedsTeXFormat{LaTeX2e}
3 \ProvidesPackage{classlist}%
4   [2008/08/11 v1.3 Record loaded classes (H0)]
5 \let\ClassList\@empty
6 \let\MainClassName\relax

Test, whether we are called before \documentclass.
7 \ifx\@classoptionslist\relax
8   \let\CL@org@fileswith@pti@ns\@fileswith@pti@ns
9   \def\@fileswith@pti@ns#1[#2]#3[#4]{%
#1:  \@clsextension
#2:  options of \documentclass/\LoadClass
#3:  class name
#4:  requested version
10    \ifx#1\@clsextension
11      \@ifl@aded#1{#3}{%
12        \PackageInfo{classlist}{%
13          Skipping class ‘#3’, because\MessageBreak
14          this class is already loaded%
15        }%
16      }{%
17        \ifundefined{MainClassName}{%
18          \def\MainClassName{#3}%
19        }{}%
20        \@temptokena\expandafter{%
21          \ClassList
22          \ClassListEntry{#3}{#2}{#4}%
23        }%
24        \edef\ClassList{\the\@temptokena}%
25      }%
26    \fi
27    \CL@org@fileswith@pti@ns{#1}[{#2}]{#3}[{#4}]%
28  }%
29  \let\@fileswith@pti@ns\@fileswith@pti@ns
```

```

30 \else
Called after \documentclass.
31 \PackageInfo{classlist}{Use \string\@filelist\space method}%
32
33 \let\ClassListEntry\relax
34 \expandafter\def\expandafter\CL@test
35 \expandafter#\expandafter1\@clsextension#2\@nil{%
36 \ifx\#2\%
Name does not contain \@clsextension
37 \else
38 \expandafter\CL@test@i\CL@entry\@nil
39 \fi
40 }%
41 \expandafter\def\expandafter\CL@test@i
42 \expandafter#\expandafter1\@clsextension#2\@nil{%
43 \ifx\#2\%
44 \@ifundefined{opt@\CL@entry}{%
45 }{%
46 \@ifundefined{MainClassName}{%
47 \let\MainClassName\CL@entry
48 }{%
49 }%
50 \edef\ClassList{%
51 \ClassList
52 \ClassListEntry{\CL@entry}{\}%
53 }%
54 }%
55 \else
Names with more than one \@clsextension are not supported.
56 \fi
57 }%
58 \@for\CL@entry:=\@filelist\do{%
59 \expandafter\expandafter\expandafter\CL@test\expandafter
60 \CL@entry\@clsextension\@nil
61 }%
62 \fi

\PrintClassListEntry
63 \providecommand*\PrintClassListEntry}[3]{%
64 \toks@{* #1}%
65 \typeout{\the\toks@}%
66 }

\PrintClassListTitle
67 \providecommand*\PrintClassListTitle}{%
68 \typeout{Class list:}%
69 }

\PrintClassList
70 \providecommand*\PrintClassList}{%
71 \begingroup
72 \let\ClassListEntry\PrintClassListEntry
73 \PrintClassListTitle
74 \ClassList
75 \endgroup
76 }

\CL@InfoEntry
77 \def\CL@InfoEntry#1#2#3{%
78 \advance\count@ by \@ne
79 \def\x{#2}%

```

```

80 \onelevel@sanitize\x
81 \edef\CL@Info{%
82   \CL@Info
83   \noexpand\MessageBreak
84   (\the\count@) %
85   #1 [\x]%
86   \ifx\#3\%
87   \else
88     \space[#3]% hash-ok
89   \fi
90 }%
91 }

92 \AtBeginDocument{%
93   \begingroup
94     \count@=\z@
95     \def\CL@Info{Class List:}%
96     \let\ClassListEntry\CL@InfoEntry
97     \ClassList
98     \let\on@line\@empty
99     \PackageInfo{classlist}{\CL@Info}%
100 \endgroup
101 }

102 \endpackage

```

## 3 Installation

### 3.1 Download

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

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

[CTAN:macros/latex/contrib/oberdiek/classlist.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: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/
```

---

<sup>1</sup><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 classlist.dtx
```

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

```
classlist.sty → tex/latex/oberdiek/classlist.sty
classlist.pdf → doc/latex/oberdiek/classlist.pdf
classlist.dtx → source/latex/oberdiek/classlist.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 classlist.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{classlist.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 $\mathrm{L}^{\mathrm{A}}\mathrm{T}_{\mathrm{E}}\mathrm{X}$` :

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

## 4 History

[2005/06/19 v1.0]

- First published version: CTAN and newsgroup `comp.text.tex`: “Re: Finding the Document Class”<sup>2</sup>

---

<sup>2</sup>Url: <http://groups.google.com/group/comp.text.tex/msg/8ee9523c2dc13666>

[2005/06/19 v1.1]

- After `\documentclass` the package looks at `\@filelist` instead of aborting with error.

[2006/02/20 v1.2]

- DTX framework.
- Fix for `\@@fileswith@pti@ns`.

[2008/08/11 v1.3]

- Code is not changed.
- URLs updated.

## 5 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
<code>\@@fileswith@pti@ns</code> .....	29	<code>\ifx</code> ..... 7, 10, 36, 43, 86
<code>\@classoptionslist</code> .....	7	
<code>\@clsextension</code> .....	10, 35, 42, 60	M
<code>\@empty</code> .....	5, 98	<code>\MainClassName</code> ..... 6, 18, 47
<code>\@filelist</code> .....	31, 58	<code>\MessageBreak</code> ..... 13, 83
<code>\@fileswith@pti@ns</code> .....	8, 9, 29	N
<code>\@for</code> .....	58	<code>\NeedsTeXFormat</code> ..... 2
<code>\@ifl@aded</code> .....	11	O
<code>\@ifundefined</code> .....	17, 44, 46	<code>\on@line</code> ..... 98
<code>\@ne</code> .....	78	P
<code>\@nil</code> .....	35, 38, 42, 60	<code>\PackageInfo</code> ..... 12, 31, 99
<code>\@onelevel@sanitize</code> .....	80	<code>\PrintClassList</code> ..... <u>70</u>
<code>\@temptokena</code> .....	20, 24	<code>\PrintClassListEntry</code> ..... <u>63</u> , 72
<code>\@</code> .....	36, 43, 86	<code>\PrintClassListTitle</code> ..... <u>67</u> , 73
		<code>\providecommand</code> ..... 63, 67, 70
A		<code>\ProvidesPackage</code> ..... 3
<code>\advance</code> .....	78	
<code>\AtBeginDocument</code> .....	92	S
		<code>\space</code> ..... 31, 88
C		T
<code>\CL@entry</code> .....	38, 44, 47, 52, 58, 60	<code>\the</code> ..... 24, 65, 84
<code>\CL@Info</code> .....	81, 82, 95, 99	<code>\toks@</code> ..... 64, 65
<code>\CL@InfoEntry</code> .....	<u>77</u> , 96	<code>\typeout</code> ..... 65, 68
<code>\CL@org@fileswith@pti@ns</code> .....	8, 27	
<code>\CL@test</code> .....	34, 59	X
<code>\CL@test@i</code> .....	38, 41	<code>\x</code> ..... 79, 80, 85
<code>\ClassList</code> ...	5, 21, 24, 50, 51, 74, 97	
<code>\ClassListEntry</code> ...	22, 33, 52, 72, 96	Z
<code>\count@</code> .....	78, 84, 94	<code>\z@</code> ..... 94
D		
<code>\do</code> .....	58	