

# The askinclude package

Pablo A. Straub

Heiko Oberdiek (maintainer)

<heiko.oberdiek at googlegmail.com>

2008/08/11 v2.1

## Abstract

This package replaces `\includeonly` by an interactive user interface.

## Contents

<b>1</b>	<b>Documentation</b>	<b>2</b>
1.1	User interface . . . . .	2
1.2	Single asking mode . . . . .	4
1.3	Regular expressions . . . . .	4
1.3.1	Method <code>makematch</code> . . . . .	4
1.3.2	Method <code>pdfmatch</code> . . . . .	5
1.4	History . . . . .	6
1.5	New features of version 2 . . . . .	6
1.6	Compatibility to version 1 . . . . .	6
<b>2</b>	<b>Implementation</b>	<b>6</b>
2.1	Header and catcodes . . . . .	6
2.2	Delay action after preamble . . . . .	7
2.3	Write entry in <code>.aux</code> file . . . . .	7
2.4	Remember <code>\include</code> files . . . . .	8
2.5	Redefine <code>\include</code> . . . . .	9
2.6	Store answer and match method in <code>.aux</code> file . . . . .	9
2.7	Support for <code>\@includeonly</code> . . . . .	9
2.8	Stop if <code>\includeonly</code> is active . . . . .	10
2.9	<code>\includeonly</code> . . . . .	10
2.10	Options . . . . .	10
2.11	Strings . . . . .	11
2.12	Set defaults . . . . .	12
2.13	Questions . . . . .	12
2.13.1	Generic question header . . . . .	12
2.13.2	Display files . . . . .	13
2.13.3	Previous answer . . . . .	13
2.13.4	Current regular expression support . . . . .	14
2.13.5	Main question text . . . . .	14
2.13.6	Main question . . . . .	14
2.14	Evaluate answer of main question . . . . .	15
2.15	Ask for each file . . . . .	18

<b>3</b>	<b>Test</b>	<b>20</b>
3.1	Included files	20
3.2	Special test setup	20
3.3	Common start	20
3.4	First run tests	23
3.5	Second run tests	28
3.6	Compatibility for changing from old to new	32
3.7	Regular expressions	33
3.8	Common end	34
<b>4</b>	<b>Installation</b>	<b>34</b>
4.1	Download	34
4.2	Bundle installation	35
4.3	Package installation	35
4.4	Refresh file name databases	36
4.5	Some details for the interested	36
<b>5</b>	<b>Acknowledgement</b>	<b>36</b>
<b>6</b>	<b>References</b>	<b>36</b>
<b>7</b>	<b>History</b>	<b>37</b>
	[1991/11/05 v0.95]	37
	[1991/11/05 v0.99]	37
	[1991/11/07 v1.0]	37
	[1991/12/13 v1.1]	37
	[1994/11/02 v1.2e]	37
	[2007/10/15 v1.0 (askinc)]	37
	[2007/10/23 v2.0]	37
	[2008/08/11 v2.1]	38
<b>8</b>	<b>Index</b>	<b>38</b>

# 1 Documentation

## 1.1 User interface

The `\include` feature of  $\text{\LaTeX}$  allows the division of the text body in separate compilation units. It is page based and stores counter settings in additional auxiliary files. `\includeonly` limits the actually included files to the ones, listed in its argument.

Example:

```

\documentclass{report}

\usepackage{askinclude}

% Don't use \include in the preamble.
% Instead \input can be used or make a package

\begin{document}
\include{title}
\include{introduction}
\include{summary}
\include{index}
\end{document}

```

Package `askinclude` adds an interactive user interface. The user is asked which files he want to include and sets `\includeonly` accordingly.

The first time the package is used the question looks like:

```
*****
*** Package askinclude Question ***
*****
```

```
Previous answer (noregexp):
[*]
```

```
Regular expressions:
--> [noregexp] disabled
[makematch] enabled, using package 'makematch'
[pdfmatch] enabled, using pdfTeX's \pdfmatch
```

```
Which files do you want to include?
[foo,bar] comma separated file or pattern list
[*] all files
[-] no files
[?] ask for each file
[] use the previous answer
```

```
\answer=chapter2,summary
```

Square brackets are used as quotation symbols, they are not part of the answer. The answer by the user follows in the last line after `\answer=`.

**[foo,bar]:** A comma separated file list (also one file only) is passed to macro `\includeonly` directly. If regular expression support is enabled, the entries in the list are interpreted as file name patterns that will be matched against the file names in `\include`, see [section 1.3](#).

**[\*]:** This is a shortcut for specifying *all* files.

**[-]:** This means *none* of the files is included.

**[?]:** The switches in single asking mode. The user will be asked at each `\include`. See [section 1.2](#).

**[]:** Just pressing return (empty string) uses the default answer, that was stored in the auxiliary file (`.aux`) of the previous `LATEX`run with package `askinclude`. Without previous runs the default is `[*]` for all files.

**[noregexp]:** Turns regular expression support off.

**[makematch]:** Enables regular expression support package `makematch` provides.

**[pdfmatch]:** It is shown, if pdf`TEX` 1.30 or later is used. It relies on an undocumented feature `\pdfmatch` that may change in future versions of pdf`TEX`.

Package `askinclude` remembers the files of `\include` in the auxiliary file and the given answer. they are presented at the next run at the start of the question:

```
*****
*** Package askinclude Question ***
*****
```

```
Files, found by previous run in \include:
(-) title
(-) introduction
(-) chapter1
(+) chapter2
(-) chapter3
(+) summary
```

```
Previous answer (noregexp):
[chapter2,summary]
```

```
...
```

(+) and (-) marks files that were and were not included in the previous run. The file list and marks are not always correct. For example, if `\include` statements are added or removed. If the method of regular expression support is different, then the marks only reflects the mode of the previous answer.

## 1.2 Single asking mode

The special answer ‘?’ trigger single asking mode, at the next `\include` the user is asked again:

```
*****
*** Package askinclude Question ***
*****

Include ‘title’? [y]es, [n]o, [A]ll, [N]one, [D]efault (n):

\answer=
```

Use the letter inside the square brackets as answer. An empty string uses the default answer, the letter in parentheses. The answers are:

[y]: The file is included.

[n]: The file is excluded.

[A]: This file and all following files are included.

[N]: This file and all following files are excluded.

[D]: Use the default answer for this and all following files.

If the file is known from the previous run, then the default is ‘y’ or ‘n’ corresponding to the mark ‘(+)’ or ‘(-)’ in the list of previous files. Otherwise ‘y’ is used as default.

## 1.3 Regular expressions

Three methods are available that can also set by package options:

**noregexp:** Regular expression support is disabled.

**makematch:** Use package `makematch`.

**pdfmatch:** Use pdfTeX’s `\pdfmatch` if available.

Without options the value remembered in the auxiliary file is used, otherwise the default is `noregexp`.

### 1.3.1 Method `makematch`

Package `makematch` (part of `qstest`), written by David Kastrup provides pattern matching based on wildcards:

[\*]: The star matches zero or more characters. Thus a lonely star machtes anything.

[!]: The exclamation mark at the begin of a pattern negates it, thus the matching files are excluded.

The pattern list is evaluated for a file name from left to right. Always a file name must match the whole pattern. The file is included if it is matched by a pattern and not excluded afterwards. Examples:

Pattern list	File names	Result
foobar	foobar,dummy	foobar
foo*	foobar,xfoobar	foobar
*foo*	xfoobar	xfoobar
f*b*r	foobar	foobar
*	any	any
!foobar	foobar,other	other
!foo*	xfoobar	xfoobar
in*,!*dex	intro,index	intro
!index,in*	intro,index	intro,index
*,!chap2	chap1,chap2,chap3	chap1,chap3

The example in the last line shows, how the behaviour of package `excludeonly` [4] can be simulated.

The documentation of package `makematch` tells the the empty pattern does not match anything. But if you answer the question with an empty string the answer of the previous run is used. Also curly braces are not reliable because the can get stripped during the processing of the answer. Using a pair of empty curly braces or the comma for two empty patterns currently works, but this is specific for the current implementation and may change in future. But the special answer ‘-’ is available.

### 1.3.2 Method `pdfmatch`

In pdfTeX 1.30 the new primitive `\pdfmatch` was introduced. It is not documented yet, because it was not decided if it remains and which implementation of regular expressions should be used. Thus the situation may change in later versions of pdfTeX. Currently (1.30 until 1.40) it uses a library for POSIX 1003.2 regular expressions, see manual page “`regex(7)`”.

If a pattern in the pattern list is invalid, the package prints a warning and removes the pattern from the list. Actually there are two warnings, the other warning is generated by pdfTeX itself.

In opposite to `makematch` the pattern only needs to match the specified part of the file name. For matching the whole name, the symbols ‘^’ and ‘\$’ for begin and end of string must be included.

Additionally an initial exclamation mark ‘!’ implements the same logic as method `makematch`. The match result is inverted, the matched file excluded.

Examples:

Pattern list	File names	Result
foobar	foobar,dummy	foobar
foo.*	foobar,xfoobar	foobar,xfoobar
^foo.*	foobar,xfoobar	foobar
foo\$	foo,foobar	foo
fo?bar	fbar,fobar,foobar	fbar,fobar
fo+bar	fbar,fobar,foobar	fobar,foobar
fo*bar	fbar,fobar,foobar	fbar,fobar,foobar
chap3	chap3,chap32	chap1,chap3,chap32
chap[13]	chap1,chap2,chap3	chap1,chap3
chap(1 3)\$	chap1,chap3,chap33	chap1,chap3
.*	any	any
!foobar	foobar,other	other
in.*,!dex	intro,index	intro
!index,in.*	intro,index	intro,index
*,!chap2	chap1,chap2,chap3	chap1,chap3
*,!bla	foo,bla,blabla	foo
*,!^bla\$	foo,bla,blabla	foo,blabla

The example in the last line shows, how the behaviour of package `excludeonly` [4]

can be simulated. As the third last line shows, that sometimes ‘`^`’ and ‘`$`’ can be omitted, but this will not work always, shown in the second last line.

## 1.4 History

Package `askinclude` was invented by Pablo A. Straub in 1991 and maintained until 1994.

In October 2007 it was removed from “TeX Live” because of the no-commercial license [1]. Herbert Voß initiated a search for the author in the newsgroup `comp.text.tex` [2] to ask him to change the license.

It seems that the author could not be reached any more. Joseph Wright suggested a reimplementaion that I (HO) have done as package `askinc`. Soon it was enhanced by feature requests.

Then Ulrike Fischer found email addresses of the original author and Jim Hefferon succeeded in contacting him. He allowed a change of the license to LPPL and the transfer of the maintainership to someone else.

Thus I merged both packages as `askinclude` in a new major version 2. The license is LPPL.

## 1.5 New features of version 2

- License is now LPPL.
- A list of `\include` files that are detected by the previous run is printed before the question.
- If the special answer ‘?’ is given, then the user is asked at each `\include`.
- Support for regular expressions.
- The remaining stuff in the `.aux` file does not cause errors, if the package is removed.
- The use of `\includeonly` before the package loading time is detected and respected by taking precedence. After the package is loaded, `\includeonly` is disabled and prints a warning.

## 1.6 Compatibility to version 1

- The user interface of version 1 with its special answers ‘\*’, ‘-’, and ‘’ remain unchanged, but it is extended by version 2 with the special answer ‘?’ and support for regular expressions can be enabled.
- As reimplementaion the internal macros have changed of course.
- `.aux` files that are written by version 1.x will work with version 2.x because `\@includeonly` is recognized.
- L<sup>A</sup>T<sub>E</sub>X 2.09: The package works as `\documentstyle` option in compatibility mode of L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>. It is possible that it will not work in pure L<sup>A</sup>T<sub>E</sub>X 2.09.

# 2 Implementation

## 2.1 Header and catcodes

```
1 <*package>
2 \NeedsTeXFormat{LaTeX2e}
3 \begingroup
4   \catcode123 1 % {
5   \catcode125 2 % }
6   \def\x{\endgroup
```

```

7    \expandafter\edef\csname AskInc@AtEnd\endcsname{%
8        \catcode35 \the\catcode35\relax
9        \catcode64 \the\catcode64\relax
10       \catcode123 \the\catcode123\relax
11       \catcode125 \the\catcode125\relax
12   }%
13 }%
14 \x
15 \catcode35 6 % #
16 \catcode64 11 % @
17 \catcode123 1 % {
18 \catcode125 2 % }
19 \def\TMP@EnsureCode#1#2{%
20     \edef\AskInc@AtEnd{%
21         \AskInc@AtEnd
22         \catcode#1 \the\catcode#1\relax
23     }%
24     \catcode#1 #2\relax
25 }
26 \TMP@EnsureCode{33}{12}% !
27 \TMP@EnsureCode{39}{12}% '
28 \TMP@EnsureCode{40}{12}% (
29 \TMP@EnsureCode{41}{12}% )
30 \TMP@EnsureCode{42}{12}% *
31 \TMP@EnsureCode{43}{12}% +
32 \TMP@EnsureCode{44}{12}% ,
33 \TMP@EnsureCode{45}{12}% -
34 \TMP@EnsureCode{46}{12}% .
35 \TMP@EnsureCode{47}{12}% /
36 \TMP@EnsureCode{58}{12}% :
37 \TMP@EnsureCode{61}{12}% =
38 \TMP@EnsureCode{62}{12}% >
39 \TMP@EnsureCode{63}{12}% ?
40 \TMP@EnsureCode{91}{12}% [
41 \TMP@EnsureCode{93}{12}% ]
42 \TMP@EnsureCode{96}{12}% '
43 \ProvidesPackage{askinclude}%
44 [2008/08/11 v2.1 Interactive asking of included files]%

```

## 2.2 Delay action after preamble

`\AskInc@AtBeginDocument` Some actions such as `.aux` file writing cannot be performed in the preamble. `\AtBeginDocument` delays such action until `\begin{document}`. However `\AtBeginDocument` must not be used after that time (`\@onlypreamble`). Thus `\AskInc@AtBeginDocument` uses `\AtBeginDocument` in the preamble and afterwards executes the code immediately.

```

45 \AtBeginDocument{%
46     \let\AskInc@AtBeginDocument\@firstofone
47 }
48 \providecommand*\AskInc@AtBeginDocument{\AtBeginDocument}

```

## 2.3 Write entry in .aux file

`\AskInc@WriteAux` The first argument is written to the main `.aux` file and the second argument is executed if writing was possible. Remember `\nofiles` prevents writing to auxiliary files.

```

49 \def\AskInc@WriteAux#1#2{%
50     \AskInc@AtBeginDocument{%
51         \if@files
52             \immediate\write\@mainaux{#1}%
53             #2%
54         \fi

```

```

55 }%
56 }

```

## 2.4 Remember \include files

```

\AskInc@CharHash
57 \edef\AskInc@CharHash{\string#}

\AskInc@WriteDefAddList
58 \def\AskInc@WriteDefAddList{%
59   \AskInc@WriteAux{%

\AskInc@AddList
60     \string\providecommand*\string\AskInc@AddList[2]{%
61       \string\ifcase\string\ifx\AskInc@CharHash1\string\undefined
62         \string\else
63         \string\ifx\AskInc@CharHash1\string\relax
64         \string\else
65         \string\ifx\AskInc@CharHash1\string\empty
66         \string\else
67         1%
68         \string\fi
69         \string\fi
70         \string\fi
71         0 %
72         \string\gdef\AskInc@CharHash1{\AskInc@CharHash2}%
73       \string\else
74       \string\g@addto@macro\AskInc@CharHash1{\AskInc@CharHash2}%
75     \string\fi
76   }%
77 }{%
78   \global\let\AskInc@WriteDefAddList\relax
79 }%
80 }

\AskInc@WriteDefAddFile
81 \def\AskInc@WriteDefAddFile{%
82   \AskInc@WriteDefAddList
83   \AskInc@WriteAux{%

\AskInc@AddFile
84     \string\providecommand\string\AskInc@AddFile{%
85       \string\AskInc@AddList\string\AskInc@Files
86     }%
87   }{%
88     \global\let\AskInc@WriteDefAddFile\relax
89   }%
90 }

\AskInc@AddFile
91 \AtBeginDocument{%
92   \global\let\AskInc@AddFile\@gobble
93 }

\AskInc@WriteAddFile
94 \def\AskInc@WriteAddFile#1{%
95   \AskInc@WriteDefAddFile
96   \AskInc@WriteAux{%
97     \string\AskInc@AddFile{#1}%
98   }{%}%
99 }

```



\AskInc@Files

```
100 \providecommand*\AskInc@Files{}
```

**Summary:** Now \AskInc@Files is always defined and either empty or contains a comma separated list of \include files of previous run.

The \include files are remembered by using \AskInc@WriteAddFile at each \include.

## 2.5 Redefine \include

\AskInc@OrgInclude

```
101 \newcommand*\AskInc@OrgInclude{}
102 \let\AskInc@OrgInclude\include
```

\AskInc@AskFile

```
103 \let\AskInc@AskFile\@gobble
```

\AskInc@MatchFile

```
104 \let\AskInc@MatchFile\@gobble
```

\include

```
105 \renewcommand*\include}[1]{%
106   \AskInc@WriteAddFile{#1}%
107   \AskInc@AskFile{#1}%
108   \AskInc@MatchFile{#1}%
109   \AskInc@OrgInclude{#1}%
110 }
```

## 2.6 Store answer and match method in .aux file

```
111 \AtEndOfPackage{%
112   \AtBeginDocument{%
113     \@ifundefined{AskInc@Answer}{-}{%
114       \AskInc@WriteAux{%
115         \string\gdef\string\AskInc@Answer{\AskInc@Answer}%
116       }{}%
117     }%
118     \@ifundefined{AskInc@MatchMethod}{-}{%
119       \AskInc@WriteAux{%
120         \string\gdef\string\AskInc@MatchMethod{\AskInc@MatchMethod}%
121       }{}%
122     }%
123   }%
124 }
```

## 2.7 Support for \@includeonly

\@includeonly Macro \@includeonly is defined for compatibility reasons only, if the .aux file was written by versions before 2 of package askinclude.

```
125 \providecommand*\@includeonly}[1]{%
126   \gdef\AskInc@Answer{#1}%
127   \ifx\AskInc@Answer\@empty
128     \gdef\AskInc@Answer{-}%
129   \fi
130   \global\let\AskInc@MatchMethod\@empty
131 }
```

## 2.8 Stop if \includeonly is active

If \includeonly is used before, the questions are skipped and the data in the main auxiliary file is preserved.

```
132 \if@partsw
133   \PackageWarningNoLine{askinclude}{}%
134     Previous \string\includeonly\space detected,\MessageBreak
135     therefore asking for files is suppressed%
136 }%
```

Dummy options.

```
137 \DeclareOption{makematch}{}%
138 \DeclareOption{pdfmatch}{}%
139 \DeclareOption{noregexp}{}%
140 \ProcessOptions*
```

Preserve \AskInc@MatchFiles.

```
141 \@ifundefined{AskInc@MatchFiles}{}%
142 }{%
143   \ifx\AskInc@MatchMethod\@empty
144     \else
145       \AskInc@WriteAux{%
146         \string\gdef\string\AskInc@MatchFiles{\AskInc@MatchFiles}%
147       }{%}%
148     \fi
149   }%
150   \AskInc@AtEnd
151   \expandafter\endinput
152 \fi
```

## 2.9 \includeonly

\AskInc@OrgIncludeOnly

```
153 \newcommand{\AskInc@OrgIncludeOnly}{}%
154 \let\AskInc@OrgIncludeOnly\includeonly
```

\includeonly

```
155 \renewcommand*{\includeonly}[1]{%
156   \PackageWarning{askinclude}{%
157     \string\includeonly\space is managed by package ‘askinclude’,%
158     \MessageBreak
159     therefore it is ignored%
160   }%
161 }
```

## 2.10 Options

\AskInc@StringNoRegExp

```
162 \def\AskInc@StringNoRegExp{noregexp}
```

\AskInc@StringMakeMatch

```
163 \def\AskInc@StringMakeMatch{makematch}
```

\AskInc@StringPdfMatch

```
164 \def\AskInc@StringPdfMatch{pdfmatch}
```

\ifAskInc@PdfMatchAvailable

```
165 \newif\ifAskInc@PdfMatchAvailable
166 \begingroup\expandafter\expandafter\expandafter\endgroup
167 \expandafter\ifx\csname pdfmatch\endcsname\relax
168 \else
169   \AskInc@PdfMatchAvailabletrue
170 \fi
```

```

171 \DeclareOption{makematch}{%
172   \let\AskInc@MatchOption\AskInc@StringMakeMatch
173 }
174 \ifAskInc@PdfMatchAvailable
175   \DeclareOption{pdfmatch}{%
176     \let\AskInc@MatchOption\AskInc@StringPdfMatch
177   }%
178 \else
179   \DeclareOption{pdfmatch}{%
180     \PackageWarningNoLine{askinclude}{%
181       Option 'pdfmatch' is not available because\MessageBreak
182       your pdfTeX does not provide \string\pdfmatch
183     }%
184   }%
185 \fi
186 \DeclareOption{noregexp}{%
187   \let\AskInc@MatchOption\@empty
188 }
189 \ProcessOptions*

```

## 2.11 Strings

\AskInc@CharAsterisk

```
190 \def\AskInc@CharAsterisk{*}
```

\AskInc@CharMinus

```
191 \def\AskInc@CharMinus{-}
```

\AskInc@CharPlus

```
192 \def\AskInc@CharPlus{+}
```

\AskInc@CharQuestion

```
193 \def\AskInc@CharQuestion{?}
```

\AskInc@CharExclam

```
194 \def\AskInc@CharExclam{!}
```

\AskInc@CharYes

```
195 \def\AskInc@CharYes{y}
```

\AskInc@CharNo

```
196 \def\AskInc@CharNo{n}
```

\AskInc@CharAll

```
197 \def\AskInc@CharAll{A}
```

\AskInc@CharNone

```
198 \def\AskInc@CharNone{N}
```

\AskInc@CharDefault

```
199 \def\AskInc@CharDefault{D}
```

## 2.12 Set defaults

\AskInc@SanitizeAnswer

```
200 \def\AskInc@SanitizeAnswer#1#2{%
201   \begingroup
202     \escapechar='\'%
203     \edef\[\{\string\[%
204     \edef\]\}\string\}%
205     \edef\({\string\}%
206     \edef\){\string\}%
207     \edef\*{\string\*}%
208     \edef\+{\string\+}%
209     \edef\?{\string\?}%
210     \edef\|{\string\|}%
211     \edef\^{\string\^}%
212     \edef\${\string\$}%
213     \edef\{\{\string\{\}%
214     \edef\}\}\string\}%
215     \edef\.\{\string\.\}%
216     \xdef#1{%
217       \expandafter\expandafter\expandafter\zap@space
218       \expandafter#2\space\@empty
219     }%
220   \endgroup
221 }

222 \AtBeginDocument{%
223   \@ifundefined{AskInc@Answer}{%
224     \global\let\AskInc@Answer\AskInc@CharAsterisk
225   }{%
226     \AskInc@SanitizeAnswer\AskInc@Answer\AskInc@Answer
227   }%
228   \global\let\AskInc@PreviousAnswer\AskInc@Answer
229   \@ifundefined{AskInc@MatchMethod}{%
230     \global\let\AskInc@MatchMethod\@empty
231   }{%
232     \global\let\AskInc@PreviousMatchMethod\AskInc@MatchMethod
233   }%
234   \let\AskInc@MatchOption\AskInc@MatchMethod
235 }{%
236   \ifx\AskInc@MatchOption\AskInc@StringPdfMatch
237     \ifAskInc@PdfMatchAvailable
238     \else
239       \let\AskInc@MatchOption\@empty
240     \fi
241   \fi
242   \@ifundefined{AskInc@MatchFiles}{%
243     \let\AskInc@MatchFiles\@empty
244   }{%
245 }
```

## 2.13 Questions

### 2.13.1 Generic question header

\AskInc@QuestionHeader

```
246 \def\AskInc@QuestionHeader{%
247   \typeout{}%
248   \typeout{*****}%
249   \typeout{*** Package askinclude Question ***}%
250   \typeout{*****}%
251   \typeout{}%
252 }
```

### 2.13.2 Display files

\AskInc@TypeoutFiles

```

253 \def\AskInc@TypeoutFiles{%
254   \global\let\AskInc@PreviousMatches\@empty
255   \ifx\AskInc@Files\@empty
256   \else
257     \typeout{Files, found by previous run in \string\include:}%
258     \@for\AskInc@File:=\AskInc@Files\do{%
259       \ifx\AskInc@Answer\AskInc@CharAsterisk
260         \let\AskInc@Mark\AskInc@CharPlus
261       \else
262         \let\AskInc@Mark\AskInc@CharMinus
263         \ifx\AskInc@Answer\AskInc@CharMinus
264         \else
265           \ifx\AskInc@MatchMethod\@empty
266             \expandafter\@firstoftwo
267           \else
268             \expandafter\@secondoftwo
269           \fi
270           {\@for\AskInc@Temp:=\AskInc@Answer}%
271           {\@for\AskInc@Temp:=\AskInc@MatchFiles}%
272         \do{%
273           \ifx\AskInc@File\AskInc@Temp
274             \let\AskInc@Mark\AskInc@CharPlus
275           \fi
276         }%
277       \fi
278     \fi
279     \typeout{ \@spaces(\AskInc@Mark) \AskInc@File}%
280     \ifx\AskInc@Mark\AskInc@CharPlus
281       \ifx\AskInc@PreviousMatches\@empty
282         \global\let\AskInc@PreviousMatches\AskInc@File
283       \else
284         \expandafter\g@addto@macro
285         \expandafter\AskInc@PreviousMatches{%
286           \expandafter,\AskInc@File
287         }%
288       \fi
289     \fi
290   }%
291   \typeout{}%
292 \fi
293 }

```

### 2.13.3 Previous answer

\AskInc@TypeoutPreviousAnswer

```

294 \def\AskInc@TypeoutPreviousAnswer{%
295   \typeout{%
296     Previous answer (
297     \ifx\AskInc@MatchMethod\@empty
298       noregexp%
299     \else
300       \AskInc@MatchMethod
301     \fi
302   ):%
303 }%
304 \typeout{ \@spaces[\AskInc@Answer]]}%
305 \typeout{}%
306 }

```

## 2.13.4 Current regular expression support

\AskInc@TypeoutMatchMethod

```

307 \def\AskInc@TypeoutMatchMethod{%
308   \typeout{Regular expressions:}%
309   \AskInc@TypeoutMethod\@empty{noregexp}{%
310     \space disabled%
311   }%
312   \AskInc@TypeoutMethod\AskInc@StringMakeMatch{makematch}{%
313     enabled, using package 'makematch'%
314   }%
315   \ifAskInc@PdfMatchAvailable
316     \AskInc@TypeoutMethod\AskInc@StringPdfMatch{pdfmatch}{%
317       \space enabled, using pdfTeX's \string\pdfmatch
318     }%
319   \fi
320   \typeout{}%
321 }
```

\AskInc@TypeoutMethod

```

322 \def\AskInc@TypeoutMethod#1#2#3{%
323   \typeout{%
324     \ifx#1\AskInc@MatchOption \space--> \else\@spaces\space\fi
325     [#2] #3% hash-ok
326   }%
327 }
```

## 2.13.5 Main question text

\AskInc@TypeoutMainQuestion

```

328 \def\AskInc@TypeoutMainQuestion{%
329   \typeout{Which files do you want to include?}%
330   \typeout{ \@spaces[foo,bar] \space\space
331     comma separated file or pattern list}%
332   \typeout{ \@spaces[*] \@spaces\@spaces all files}%
333   \typeout{ \@spaces[-] \@spaces\@spaces no files}%
334   \typeout{ \@spaces[?] \@spaces\@spaces ask for each file}%
335 }
```

## 2.13.6 Main question

\AskInc@MainQuestion

```

336 \def\AskInc@MainQuestion{%
337   \AskInc@QuestionHeader
338   \AskInc@TypeoutFiles
339   \AskInc@TypeoutPreviousAnswer
340   \AskInc@TypeoutMatchMethod
341   \AskInc@TypeoutMainQuestion
342   \beginngroup
343     \@makeother\*%
344     \@makeother\-%
345     \@makeother\?%
346     \@makeother\,%
347     \@makeother\!%
348     \@makeother\^%
349     \@makeother\$%
350     \@makeother\(%
351     \@makeother\)%
352     \@makeother\+%
353     \@makeother\[%
354     \@makeother\]%

```

```

355 \makeother\|
356 \makeother\|
357 \makeother\:%
358 \makeother\.%
359 \typein[\answer]%
360 { \spaces[] \space\spaces\spaces use the previous answer}%
361 \typeout{}%
362 \AskInc@SanitizeAnswer\AskInc@@Answer\answer
363 \endgroup
364 \let\AskInc@Next\AskInc@MainQuestion
365 \ifx\AskInc@@Answer\AskInc@StringNoRegExp
366 \let\AskInc@MatchOption\@empty
367 \else
368 \ifx\AskInc@@Answer\AskInc@StringMakeMatch
369 \let\AskInc@MatchOption\AskInc@StringMakeMatch
370 \else
371 \ifx\AskInc@@Answer\AskInc@StringPdfMatch
372 \ifAskInc@PdfMatchAvailable
373 \let\AskInc@MatchOption\AskInc@StringPdfMatch
374 \else
375 \PackageWarning{askinclude}{%
376 Method 'pdfmatch' is not available because\MessageBreak
377 your pdfTeX does not provide \string\pdfmatch
378 }%
379 \let\AskInc@MatchOption\@empty
380 \fi
381 \else
382 \ifx\AskInc@@Answer\@empty
383 \global\let\AskInc@@Answer\AskInc@Answer
384 \fi
385 \let\AskInc@Next\AskInc@MainAnswer
386 \fi
387 \fi
388 \fi
389 \AskInc@Next
390 }

```

## 2.14 Evaluate answer of main question

\AskInc@MainAnswer

```

391 \def\AskInc@MainAnswer{%
392 \ifx\AskInc@@Answer\AskInc@CharQuestion
393 \global\let\AskInc@Answer\@empty
394 \global\let\AskInc@MatchMethod\@empty
395 \global\let\AskInc@AskFile\AskInc@DoAskFile
396 \AskInc@OrgIncludeOnly{}%
397 \else
398 \global\let\AskInc@Answer\AskInc@@Answer
399 \ifx\AskInc@@Answer\AskInc@CharAsterisk
400 \global\let\AskInc@MatchMethod\@empty
401 \else
402 \ifx\AskInc@@Answer\AskInc@CharMinus
403 \global\let\AskInc@MatchMethod\@empty
404 \AskInc@OrgIncludeOnly{}%
405 \else
406 \global\let\AskInc@MatchMethod\AskInc@MatchOption
407 \ifx\AskInc@MatchOption\@empty
408 \expandafter\AskInc@OrgIncludeOnly\expandafter{%
409 \AskInc@Answer
410 }%
411 \else
412 \AskInc@MatchInit

```

```

413         \AskInc@OrgIncludeOnly{%
414         \fi
415     \fi
416 \fi
417 \fi
418 \let\AskInc@Include\AskInc@DoInclude
419 }

\AskInc@MatchInit

420 \def\AskInc@MatchInit{%
421     \RequirePackage{kvsetkeys}[2007/09/29]%
422     \expandafter\comma@normalize\expandafter{\AskInc@Answer}%
423     \global\let\AskInc@MatchList\comma@list

\AskInc@MatchFile

424 \def\AskInc@MatchFile##1{%
425     \AskInc@DoMatch{##1}%
426     \AskInc@WriteAddMatch{##1}%
427     \ifx\@partlist\@empty
428         \gdef\@partlist{##1}%
429     \else
430         \g@addto@macro\@partlist{,##1}%
431     \fi
432 }%
433 }%

434 \ifx\AskInc@MatchMethod\AskInc@StringMakeMatch
435     \AskInc@MakeMatchInit
436 \else
437     \AskInc@PdfMatchInit
438 \fi
439 \let\AskInc@MatchInit\@undefined
440 }

\AskInc@MakeMatchInit

441 \def\AskInc@MakeMatchInit{%
442     \RequirePackage{makematch}%

\AskInc@DoMatch

443 \def\AskInc@DoMatch##1{%
444     \def\AskInc@Temp{%
445         \MakeMatcher[,]\AskInc@Temp
446     }%
447     \expandafter\AskInc@Temp\expandafter{\AskInc@MatchList}%
448     \AskInc@Temp{{##1}}\@firstofone\@gobble
449 }%

450 \let\AskInc@MakeMatchInit\@undefined
451 }

\AskInc@PdfMatchInit

452 \def\AskInc@PdfMatchInit{%

\ifAskInc@Found

453 \newif\ifAskInc@Found

\AskInc@DoMatch

454 \def\AskInc@DoMatch##1{%
455     \AskInc@Foundfalse
456     \expandafter\comma@parse@normalized\expandafter
457     {\AskInc@MatchList}{\AskInc@CheckMatch{##1}}%

```



```

458     \ifAskInc@Found
459     \expandafter\@firstofone
460     \else
461     \expandafter\@gobble
462     \fi
463 }%

\AskInc@CheckMatch

464 \def\AskInc@CheckMatch##1##2{%
465     \AskInc@@CheckMatch##2\@empty\@nil{##1}%
466 }%

\AskInc@@CheckMatch

467 \def\AskInc@@CheckMatch##1##2\@nil##3{%
468     \def\AskInc@Temp{##1}%
469     \ifx\AskInc@Temp\AskInc@CharExclam
470     \ifcase\pdfmatch{##2}{##3}=1 %
471     \or
472     \AskInc@Foundfalse
473     \else
474     \AskInc@InvalidPattern{##2}{##1##2}%
475     \fi
476     \else
477     \ifcase\pdfmatch{##1##2}{##3} %
478     \or
479     \AskInc@Foundtrue
480     \else
481     \AskInc@InvalidPattern{##1##2}{##1##2}%
482     \fi
483     \fi
484 }%

\AskInc@InvalidPattern Prints a warning with the regular expression and removes it from the list to prevent
further warnings for the same expression.

485 \def\AskInc@InvalidPattern##1##2{%
486     \edef\AskInc@Temp{##1}%
487     \PackageWarningNoLine{askinclude}{%
488         Invalid pattern '\AskInc@Temp' for \string\pdfmatch
489     }%
490     \edef\AskInc@Temp{##2}%
491     \global\expandafter\let\expandafter\AskInc@MatchList
492     \expandafter\@empty
493     \expandafter\comma@parse@normalized\expandafter
494     {\AskInc@MatchList}\AskInc@FilterPattern
495 }%

\AskInc@FilterPattern

496 \def\AskInc@FilterPattern##1{%
497     \edef\AskInc@Pattern{##1}%
498     \ifx\AskInc@Temp\AskInc@Pattern
499     \else
500     \ifx\AskInc@MatchList\@empty
501     \global\let\AskInc@MatchList\AskInc@Pattern
502     \else
503     \expandafter\g@addto@macro\expandafter\AskInc@MatchList{%
504     \expandafter,\AskInc@Pattern
505     }%
506     \fi
507     \fi
508 }%

509 \let\AskInc@PdfMatchInit\@undefined
510 }

```

\AskInc@WriteDefAddMatch

```
511 \def\AskInc@WriteDefAddMatch{%
512   \AskInc@WriteDefAddList
513   \AskInc@WriteAux{%
```

\AskInc@AddMatch

```
514   \string\providecommand\string\AskInc@AddMatch{%
515     \string\AskInc@AddList\string\AskInc@MatchFiles
516   }%

517 }{%
518   \global\let\AskInc@WriteDefAddMatch\relax
519 }%
520 }
```

\AskInc@AddMatch

```
521 \AtBeginDocument{%
522   \global\let\AskInc@AddMatch\@gobble
523 }
```

\AskInc@WriteAddMatch

```
524 \def\AskInc@WriteAddMatch#1{%
525   \AskInc@WriteDefAddMatch
526   \AskInc@WriteAux{%
527     \string\AskInc@AddMatch{#1}%
528   }{%
529 }
```

\AskInc@SaveAnswer

```
530 \def\AskInc@SaveAnswer{%
531   \AskInc@WriteAux{%
532     \string\gdef\string\AskInc@Answer{\AskInc@Answer}%
533   }{%
534 }
```

## 2.15 Ask for each file

\AskInc@PreviousAnswer

```
535 \@ifundefined{AskInc@PreviousAnswer}{%
536   \let\AskInc@PreviousAnswer\AskInc@CharAsterisk
537 }{}
```

\AskInc@PreviousMatchMethod

```
538 \@ifundefined{AskInc@PreviousMatchMethod}{%
539   \let\AskInc@PreviousMatchMethod\@empty
540 }{}
```

\AskInc@DoAskFile

```
541 \def\AskInc@DoAskFile#1{%
542   \edef\AskInc@File{#1}%
543   \ifx\AskInc@PreviousAnswer\AskInc@CharAsterisk
544     \let\AskInc@Default\AskInc@CharYes
545   \else
546     \let\AskInc@Default\AskInc@CharNo
547     \ifx\AskInc@PreviousAnswer\AskInc@CharMinus
548       \else
549         \ifx\AskInc@PreviousMatchMethod\@empty
550           \expandafter\@firstoftwo
551         \else
552           \expandafter\@secondoftwo
```

```

553     \fi
554     {\@for\AskInc@Temp:=\AskInc@PreviousAnswer}%
555     {\@for\AskInc@Temp:=\AskInc@MatchFiles}%
556     \do{%
557         \ifx\AskInc@Temp\AskInc@File
558             \let\AskInc@Default\AskInc@CharYes
559         \fi
560     }%
561 \fi
562 \ifx\AskInc@Default\AskInc@CharNo
563     \let\AskInc@Default\AskInc@CharYes
564     \@for\AskInc@Temp:=\AskInc@Files\do{%
565         \ifx\AskInc@Temp\AskInc@File
566             \let\AskInc@Default\AskInc@CharNo
567         \fi
568     }%
569 \fi
570 \fi
571 \let\AskInc@@QuestionHeader\AskInc@QuestionHeader
572 \AskInc@AskQuestion
573 \ifx\AskInc@AskResult\AskInc@CharYes
574     \ifx\AskInc@Answer\@empty
575         \gdef\AskInc@Answer{#1}%
576     \else
577         \g@addto@macro\AskInc@Answer{, #1}%
578     \fi
579     \AskInc@SaveAnswer
580     \let\@partlist\AskInc@Answer
581 \fi
582 }

```

\AskInc@AskQuestion

```

583 \def\AskInc@AskQuestion{%
584     \let\AskInc@Temp\answer
585     \AskInc@@QuestionHeader
586     \let\AskInc@@QuestionHeader\relax
587     \typein[\answer]{%
588         Include '\AskInc@File'? [y]es, [n]o, [A]ll, [N]one, [D]efault %
589         (\AskInc@Default):%
590     }%
591     \typeout{}%
592     \let\AskInc@AskResult\answer
593     \let\answer\AskInc@Temp
594     \ifx\AskInc@AskResult\@empty
595         \let\AskInc@AskResult\AskInc@Default
596         \let\AskInc@Next\relax
597     \else
598         \ifx\AskInc@AskResult\AskInc@CharYes
599             \let\AskInc@Next\relax
600         \else
601             \ifx\AskInc@AskResult\AskInc@CharNo
602                 \let\AskInc@Next\relax
603             \else
604                 \ifx\AskInc@AskResult\AskInc@CharAll
605                     \gdef\AskInc@AskQuestion{%
606                         \let\AskInc@AskResult\AskInc@CharYes
607                     }%
608                 \else
609                     \ifx\AskInc@AskResult\AskInc@CharNone
610                         \gdef\AskInc@AskQuestion{%
611                             \let\AskInc@AskResult\AskInc@CharNo
612                         }%
613                     \else

```

```

614         \ifx\AskInc@AskResult\AskInc@CharDefault
615         \gdef\AskInc@AskQuestion{%
616         \let\AskInc@AskResult\AskInc@Default
617         }%
618     \else
619         \PackageWarningNoLine{askinclude}{%
620         Invalid answer [\AskInc@AskResult].\MessageBreak
621         Answer must be one letter out of [ynAND]%
622         }%
623     \fi
624 \fi
625 \fi
626 \let\AskInc@Next\AskInc@AskQuestion
627 \fi
628 \fi
629 \fi
630 \AskInc@Next
631 }

632 \AtBeginDocument{\AskInc@MainQuestion}

633 \AskInc@AtEnd
634 \end{package}

```

## 3 Test

### 3.1 Included files

```

635 <*a>
636 \chapter{Chapter A}
637 \let\filea=Y
638 </a>

639 <*b>
640 \chapter{Chapter B}
641 \let\fileb=Y
642 </b>

643 <*c>
644 \chapter{Chapter C}
645 \let\filec=Y
646 </c>

```

### 3.2 Special test setup

```

647 <*test18>
648 \includeonly{askinclude-b}
649 </test18>

650 <*test20 | test21 | test22>
651 \let\AtBeginDocumentMode=N%
652 </test20 | test21 | test22>

```

### 3.3 Common start

```

653 <*test0>
654 \ifx\CompatibilityMode Y%
655 \documentstyle[filecontents,qstest,askinclude]{book}%
656 \else
657 \documentclass{book}%
658 \usepackage{filecontents}%
659 \usepackage{qstest}%
660 <*test1>
661 \makeatletter
662 \def\RestoreCatcodes{}%
663 \count@=0 %

```

```

664 \loop
665   \edef\RestoreCatcodes{%
666     \RestoreCatcodes
667     \catcode\the\count@=\the\catcode\count@\relax
668   }%
669 \ifnum\count@<255 %
670   \advance\count@\@ne
671 \repeat
672 \def\RangeCatcodeInvalid#1#2{%
673   \count@=#1\relax
674   \loop
675     \catcode\count@=15 %
676   \ifnum\count@<#2\relax
677     \advance\count@\@ne
678   \repeat
679 }%
680 \def\Test#1\EndTest{%
681   \RangeCatcodeInvalid{0}{47}%
682   \RangeCatcodeInvalid{58}{64}%
683   \RangeCatcodeInvalid{91}{96}%
684   \RangeCatcodeInvalid{123}{255}%
685   \catcode'\@=12 %
686   \catcode'\=0 %
687   \catcode'\{=1 %
688   \catcode'\}=2 %
689   \catcode'\#=6 %
690   \catcode'\[=12 %
691   \catcode'\]=12 %
692   \catcode'\%=14 %
693   \catcode'\ =10 %
694   \catcode13=5 %
695   #1\relax
696   \RestoreCatcodes
697   \makeatother
698 }%
699 \Test
700 </test1>
701 <!*test1>
702 \ifx\AtBeginDocumentMode Y%
703   \AtBeginDocument{%
704     \usepackage{askinclud}[2008/08/11]%
705   }%
706 \else
707 </!test1>
708   \usepackage{askinclud}[2008/08/11]%
709 <!*test1>
710 \fi
711 </!test1>
712 <*test1>
713 \EndTest
714 </test1>
715 \fi
716 \IncludeTests{*}
717 \LogTests{log}{*}{*}
718 \makeatletter
719 \let\org@typein\typein
720 \def\typein[#1]#2{%
721   \ifx\@Stack\@StackEmpty
722     \begin{qstest}{StackNonEmpty}{StackNonEmpty}%
723       \Expect{Error: empty stack}{}%
724     \end{qstest}%
725     \org@typein[#1]{#2}%

```

```

726 \else
727   \edef#1{%
728     \unexpanded\expandafter\expandafter\expandafter{%
729       \expandafter\@car\@Stack\@nil
730     }%
731   }%
732   \xdef\@Stack{%
733     \unexpanded\expandafter\expandafter\expandafter{%
734       \expandafter\@cdr\@Stack\@nil
735     }%
736   }%
737   \typeout{%
738     \typeout{\string#1=#1}%
739   \fi
740 }
741 \def\@StackEmpty{\BOTTOM}
742 \let\@Stack\@StackEmpty
743 \newcommand*\@Stack[1]{%
744   \def\@Stack{#1\BOTTOM}%
745 }
746 \AtEndDocument{%
747   \begin{qstest}{StackUsed}{StackUsed}%
748     \Expect*{%
749       \ifx\@Stack\@StackEmpty
750       \else
751         Error: Unused stack entries%
752       \fi
753     }{}%
754   \end{qstest}%
755 }
756 \newcommand*\@CheckInclude[2]{%
757   \include{askinclude-#1}%
758   \begin{qstest}{CheckInclude-#1}{CheckInclude-#1}%
759     \Expect*{%
760       \expandafter\ifx\csname file#1\endcsname Ytrue\else false\fi
761     }{#2}%
762   \end{qstest}%
763 }
764 \def\@EndChecks{}
765 \AtBeginDocument{%
766   \AtEndDocument{%
767     \if@filesw
768       \immediate\write\@mainaux{\string\@EndChecks}%
769     \fi
770   }%
771 }
772 \newcommand*\@CheckAnswer[1]{%
773   \gdef\AnswerText{#1}%
774   \g@addto@macro\@EndChecks{\@CheckAnswer}%
775 }
776 \def\@CheckAnswer{%
777   \begin{qstest}{CheckAnswer}{CheckAnswer}%
778     \Expect*\unexpanded\expandafter{\AskInc@Answer}}%
779     *\unexpanded\expandafter{\AnswerText}}%
780   \end{qstest}%
781 }
782 \newcommand*\@CheckMatchMethod[1]{%
783   \gdef\MatchMethodText{#1}%
784   \g@addto@macro\@EndChecks{\@CheckMatchMethod}%
785 }
786 \def\@CheckMatchMethod{%
787   \begin{qstest}{CheckMatchMethod}{CheckMatchMethod}%

```

```

788 \Expect*{\unexpanded\expandafter{\AskInc@MatchMethod}}%
789 *{\unexpanded\expandafter{\MatchMethodText}}%
790 \end{qstest}%
791 }
792 \def\@CheckAux{%
793 \begin{qstest}{\CheckAux}{\CheckAux}%
794 \def\GetFileContents##1{%
795 \pdfunescapehex{%
796 \pdffiledump length\pdffilesize{##1}{##1}%
797 }%
798 }%
799 \Expect*{\GetFileContents{\jobname.aux}}%
800 *{\GetFileContents{\jobname-aux.aux}}%
801 \end{qstest}%
802 }
803 \g@addto@macro\@EndChecks{\@CheckAux}
804 \let\org@filecontents\filecontents
805 \def\filecontents{%
806 \catcode'\|=0 %
807 \org@filecontents
808 }
809 \begingroup
810 \endlinechar=-1 %
811 \catcode'\|=0 %
812 \@makeother\#%
813 \@makeother\|%
814 |gdef|AddList{%
815 \providecommand\AskInc@AddList[2]{%
816 \ifcase\ifx#1\@undefined
817 \else\ifx#1\relax
818 \else\ifx#1\@empty
819 \else1\fi\fi\fi0 %
820 \gdef#1{#2}%
821 \else
822 \g@addto@macro#1{,#2}%
823 \fi
824 }%
825 }%
826 |gdef|AddFile{%
827 \providecommand\AskInc@AddFile{%
828 \AskInc@AddList\AskInc@Files
829 }%
830 }%
831 |gdef|AddMatch{%
832 \providecommand\AskInc@AddMatch{%
833 \AskInc@AddList\AskInc@MatchFiles
834 }%
835 }%
836 |endgroup
837 \end{test0}

```

### 3.4 First run tests

```

838 \test1
839 \Stack{}
840 \CheckAnswer{*}
841 \begin{filecontents*}{\jobname.aux}
842 \relax
843 \end{filecontents*}
844 \begin{filecontents*}{\jobname-aux.aux}
845 \relax\space
846 \gdef\AskInc@Answer{*}
847 \gdef\AskInc@MatchMethod{}

```

```

848 |AddList
849 |AddFile
850 \AskInc@AddFile{askinclude-a}
851 \@input{askinclude-a.aux}
852 \AskInc@AddFile{askinclude-b}
853 \@input{askinclude-b.aux}
854 \AskInc@AddFile{askinclude-c}
855 \@input{askinclude-c.aux}
856 \@EndChecks
857 \end{filecontents*}
858 \begin{document}
859 \CheckInclude{a}{true}
860 \CheckInclude{b}{true}
861 \CheckInclude{c}{true}
862 </test1>

863 <*test2>
864 \Stack{*}
865 \CheckAnswer{*}
866 \begin{filecontents*}{\jobname.aux}
867 \relax
868 \end{filecontents*}
869 \begin{filecontents*}{\jobname-aux.aux}
870 \relax|space
871 \gdef\AskInc@Answer{*}
872 \gdef\AskInc@MatchMethod{}
873 |AddList
874 |AddFile
875 \AskInc@AddFile{askinclude-a}
876 \@input{askinclude-a.aux}
877 \AskInc@AddFile{askinclude-b}
878 \@input{askinclude-b.aux}
879 \AskInc@AddFile{askinclude-c}
880 \@input{askinclude-c.aux}
881 \@EndChecks
882 \end{filecontents*}
883 \begin{document}
884 \CheckInclude{a}{true}
885 \CheckInclude{b}{true}
886 \CheckInclude{c}{true}
887 </test2>

888 <*test3>
889 \Stack{-}%
890 \CheckAnswer{-}
891 \begin{filecontents*}{\jobname.aux}
892 \relax
893 \end{filecontents*}
894 \begin{filecontents*}{\jobname-aux.aux}
895 \relax|space
896 \gdef\AskInc@Answer{-}
897 \gdef\AskInc@MatchMethod{}
898 |AddList
899 |AddFile
900 \AskInc@AddFile{askinclude-a}
901 \@input{askinclude-a.aux}
902 \AskInc@AddFile{askinclude-b}
903 \@input{askinclude-b.aux}
904 \AskInc@AddFile{askinclude-c}
905 \@input{askinclude-c.aux}
906 \@EndChecks
907 \end{filecontents*}
908 \begin{document}
909 \CheckInclude{a}{false}

```



```

910 \CheckInclude{b}{false}
911 \CheckInclude{c}{false}
912 \test3}
913 {*test4}
914 \Stack{{askinclude-b}}
915 \CheckAnswer{askinclude-b}
916 \begin{filecontents*}{\jobname.aux}
917 \relax
918 \end{filecontents*}
919 \begin{filecontents*}{\jobname-aux.aux}
920 \relax|space
921 \gdef\AskInc@Answer{askinclude-b}
922 \gdef\AskInc@MatchMethod{}
923 |AddList
924 |AddFile
925 \AskInc@AddFile{askinclude-a}
926 \@input{askinclude-a.aux}
927 \AskInc@AddFile{askinclude-b}
928 \@input{askinclude-b.aux}
929 \AskInc@AddFile{askinclude-c}
930 \@input{askinclude-c.aux}
931 \@EndChecks
932 \end{filecontents*}
933 \begin{document}
934 \CheckInclude{a}{false}
935 \CheckInclude{b}{true}
936 \CheckInclude{c}{false}
937 \test4}
938 {*test5}
939 \Stack{{askinclude-a,askinclude-c}}
940 \CheckAnswer{askinclude-a,askinclude-c}
941 \begin{filecontents*}{\jobname.aux}
942 \relax
943 \end{filecontents*}
944 \begin{filecontents*}{\jobname-aux.aux}
945 \relax|space
946 \gdef\AskInc@Answer{askinclude-a,askinclude-c}
947 \gdef\AskInc@MatchMethod{}
948 |AddList
949 |AddFile
950 \AskInc@AddFile{askinclude-a}
951 \@input{askinclude-a.aux}
952 \AskInc@AddFile{askinclude-b}
953 \@input{askinclude-b.aux}
954 \AskInc@AddFile{askinclude-c}
955 \@input{askinclude-c.aux}
956 \@EndChecks
957 \end{filecontents*}
958 \begin{document}
959 \CheckInclude{a}{true}
960 \CheckInclude{b}{false}
961 \CheckInclude{c}{true}
962 \test5}
963 {*test6}
964 \Stack{?yny}
965 \CheckAnswer{askinclude-a,askinclude-c}
966 \begin{filecontents*}{\jobname.aux}
967 \relax
968 \end{filecontents*}
969 \begin{filecontents*}{\jobname-aux.aux}
970 \relax|space
971 \gdef\AskInc@Answer{}

```

```

972 \gdef\AskInc@MatchMethod{}
973 |AddList
974 |AddFile
975 \AskInc@AddFile{askinclude-a}
976 \gdef\AskInc@Answer{askinclude-a}
977 \@input{askinclude-a.aux}
978 \AskInc@AddFile{askinclude-b}
979 \@input{askinclude-b.aux}
980 \AskInc@AddFile{askinclude-c}
981 \gdef\AskInc@Answer{askinclude-a,askinclude-c}
982 \@input{askinclude-c.aux}
983 \@EndChecks
984 \end{filecontents*}
985 \begin{document}
986 \CheckInclude{a}{true}
987 \CheckInclude{b}{false}
988 \CheckInclude{c}{true}
989 </test6>

990 <*test7>
991 \Stack{?yA}
992 \CheckAnswer{askinclude-a,askinclude-b,askinclude-c}
993 \begin{filecontents*}{\jobname.aux}
994 \relax
995 \end{filecontents*}
996 \begin{filecontents*}{\jobname-aux.aux}
997 \relax|space
998 \gdef\AskInc@Answer{}
999 \gdef\AskInc@MatchMethod{}
1000 |AddList
1001 |AddFile
1002 \AskInc@AddFile{askinclude-a}
1003 \gdef\AskInc@Answer{askinclude-a}
1004 \@input{askinclude-a.aux}
1005 \AskInc@AddFile{askinclude-b}
1006 \gdef\AskInc@Answer{askinclude-a,askinclude-b}
1007 \@input{askinclude-b.aux}
1008 \AskInc@AddFile{askinclude-c}
1009 \gdef\AskInc@Answer{askinclude-a,askinclude-b,askinclude-c}
1010 \@input{askinclude-c.aux}
1011 \@EndChecks
1012 \end{filecontents*}
1013 \begin{document}
1014 \CheckInclude{a}{true}
1015 \CheckInclude{b}{true}
1016 \CheckInclude{c}{true}
1017 </test7>

1018 <*test8>
1019 \Stack{?yD}
1020 \CheckAnswer{askinclude-a,askinclude-b,askinclude-c}
1021 \begin{filecontents*}{\jobname.aux}
1022 \relax
1023 \end{filecontents*}
1024 \begin{filecontents*}{\jobname-aux.aux}
1025 \relax|space
1026 \gdef\AskInc@Answer{}
1027 \gdef\AskInc@MatchMethod{}
1028 |AddList
1029 |AddFile
1030 \AskInc@AddFile{askinclude-a}
1031 \gdef\AskInc@Answer{askinclude-a}
1032 \@input{askinclude-a.aux}
1033 \AskInc@AddFile{askinclude-b}

```

```

1034 \gdef\AskInc@Answer{askinclude-a,askinclude-b}
1035 \@input{askinclude-b.aux}
1036 \AskInc@AddFile{askinclude-c}
1037 \gdef\AskInc@Answer{askinclude-a,askinclude-b,askinclude-c}
1038 \@input{askinclude-c.aux}
1039 \@EndChecks
1040 \end{filecontents*}
1041 \begin{document}
1042 \CheckInclude{a}{true}
1043 \CheckInclude{b}{true}
1044 \CheckInclude{c}{true}
1045 </test8>

1046 <*test9>
1047 \Stack{?yN}
1048 \CheckAnswer{askinclude-a}
1049 \begin{filecontents*}{\jobname.aux}
1050 \relax
1051 \end{filecontents*}
1052 \begin{filecontents*}{\jobname-aux.aux}
1053 \relax|space
1054 \gdef\AskInc@Answer{}
1055 \gdef\AskInc@MatchMethod{}
1056 |AddList
1057 |AddFile
1058 \AskInc@AddFile{askinclude-a}
1059 \gdef\AskInc@Answer{askinclude-a}
1060 \@input{askinclude-a.aux}
1061 \AskInc@AddFile{askinclude-b}
1062 \@input{askinclude-b.aux}
1063 \AskInc@AddFile{askinclude-c}
1064 \@input{askinclude-c.aux}
1065 \@EndChecks
1066 \end{filecontents*}
1067 \begin{document}
1068 \CheckInclude{a}{true}
1069 \CheckInclude{b}{false}
1070 \CheckInclude{c}{false}
1071 </test9>

1072 <*test10>
1073 \Stack{?{}N}
1074 \CheckAnswer{askinclude-a}
1075 \begin{filecontents*}{\jobname.aux}
1076 \relax
1077 \end{filecontents*}
1078 \begin{filecontents*}{\jobname-aux.aux}
1079 \relax|space
1080 \gdef\AskInc@Answer{}
1081 \gdef\AskInc@MatchMethod{}
1082 |AddList
1083 |AddFile
1084 \AskInc@AddFile{askinclude-a}
1085 \gdef\AskInc@Answer{askinclude-a}
1086 \@input{askinclude-a.aux}
1087 \AskInc@AddFile{askinclude-b}
1088 \@input{askinclude-b.aux}
1089 \AskInc@AddFile{askinclude-c}
1090 \@input{askinclude-c.aux}
1091 \@EndChecks
1092 \end{filecontents*}
1093 \begin{document}
1094 \CheckInclude{a}{true}
1095 \CheckInclude{b}{false}

```

```

1096 \CheckInclude{c}{false}
1097 </test10>

1098 <*test11>
1099 \Stack{?xyxxN}
1100 \CheckAnswer{askinclude-a}
1101 \begin{filecontents*}{\jobname.aux}
1102 \relax
1103 \end{filecontents*}
1104 \begin{filecontents*}{\jobname-aux.aux}
1105 \relax|space
1106 \gdef\AskInc@Answer{}
1107 \gdef\AskInc@MatchMethod{}
1108 |AddList
1109 |AddFile
1110 \AskInc@AddFile{askinclude-a}
1111 \gdef\AskInc@Answer{askinclude-a}
1112 \@input{askinclude-a.aux}
1113 \AskInc@AddFile{askinclude-b}
1114 \@input{askinclude-b.aux}
1115 \AskInc@AddFile{askinclude-c}
1116 \@input{askinclude-c.aux}
1117 \@EndChecks
1118 \end{filecontents*}
1119 \begin{document}
1120 \CheckInclude{a}{true}
1121 \CheckInclude{b}{false}
1122 \CheckInclude{c}{false}
1123 </test11>

```

### 3.5 Second run tests

```

1124 <*test12>
1125 \Stack{{}}
1126 \CheckAnswer{askinclude-b}
1127 \begin{filecontents*}{\jobname.aux}
1128 \gdef\AskInc@Answer{askinclude-b}
1129 \gdef\AskInc@MatchMethod{}
1130 |AddList
1131 |AddFile
1132 \AskInc@AddFile{askinclude-a}
1133 \AskInc@AddFile{askinclude-b}
1134 \AskInc@AddFile{askinclude-c}
1135 \end{filecontents*}
1136 \begin{filecontents*}{\jobname-aux.aux}
1137 \relax|space
1138 \gdef\AskInc@Answer{askinclude-b}
1139 \gdef\AskInc@MatchMethod{}
1140 |AddList
1141 |AddFile
1142 \AskInc@AddFile{askinclude-a}
1143 \@input{askinclude-a.aux}
1144 \AskInc@AddFile{askinclude-b}
1145 \@input{askinclude-b.aux}
1146 \AskInc@AddFile{askinclude-c}
1147 \@input{askinclude-c.aux}
1148 \@EndChecks
1149 \end{filecontents*}
1150 \begin{document}
1151 \CheckInclude{a}{false}
1152 \CheckInclude{b}{true}
1153 \CheckInclude{c}{false}
1154 </test12>
1155 <*test13>

```

```

1156 \Stack{[]}
1157 \CheckAnswer{askinclude-b}
1158 \begin{filecontents*}{\jobname.aux}
1159 \gdef\AskInc@Answer{askinclude-b}
1160 \gdef\AskInc@MatchMethod{}
1161 |AddList
1162 |AddFile
1163 \AskInc@AddFile{askinclude-a}
1164 \AskInc@AddFile{askinclude-b}
1165 \end{filecontents*}
1166 \begin{filecontents*}{\jobname-aux.aux}
1167 \relax|space
1168 \gdef\AskInc@Answer{askinclude-b}
1169 \gdef\AskInc@MatchMethod{}
1170 |AddList
1171 |AddFile
1172 \AskInc@AddFile{askinclude-a}
1173 \@input{askinclude-a.aux}
1174 \AskInc@AddFile{askinclude-b}
1175 \@input{askinclude-b.aux}
1176 \AskInc@AddFile{askinclude-c}
1177 \@input{askinclude-c.aux}
1178 \@EndChecks
1179 \end{filecontents*}
1180 \begin{document}
1181 \CheckInclude{a}{false}
1182 \CheckInclude{b}{true}
1183 \CheckInclude{c}{false}
1184 </test13>

1185 <*test14>
1186 \Stack{[]}
1187 \CheckAnswer{*}
1188 \begin{filecontents*}{\jobname.aux}
1189 \gdef\AskInc@Answer{*}
1190 \gdef\AskInc@MatchMethod{}
1191 |AddList
1192 |AddFile
1193 \AskInc@AddFile{askinclude-a}
1194 \AskInc@AddFile{askinclude-b}
1195 \end{filecontents*}
1196 \begin{filecontents*}{\jobname-aux.aux}
1197 \relax|space
1198 \gdef\AskInc@Answer{*}
1199 \gdef\AskInc@MatchMethod{}
1200 |AddList
1201 |AddFile
1202 \AskInc@AddFile{askinclude-a}
1203 \@input{askinclude-a.aux}
1204 \AskInc@AddFile{askinclude-b}
1205 \@input{askinclude-b.aux}
1206 \AskInc@AddFile{askinclude-c}
1207 \@input{askinclude-c.aux}
1208 \@EndChecks
1209 \end{filecontents*}
1210 \begin{document}
1211 \CheckInclude{a}{true}
1212 \CheckInclude{b}{true}
1213 \CheckInclude{c}{true}
1214 </test14>

1215 <*test15>
1216 \Stack{?D}
1217 \CheckAnswer{askinclude-a,askinclude-c}

```

```

1218 \begin{filecontents*}{\jobname.aux}
1219 \gdef\AskInc@Answer{askinclude-a}
1220 \gdef\AskInc@MatchMethod{}
1221 |AddList
1222 |AddFile
1223 \AskInc@AddFile{askinclude-a}
1224 \AskInc@AddFile{askinclude-b}
1225 \end{filecontents*}
1226 \begin{filecontents*}{\jobname-aux.aux}
1227 \relax|space
1228 \gdef\AskInc@Answer{}
1229 \gdef\AskInc@MatchMethod{}
1230 |AddList
1231 |AddFile
1232 \AskInc@AddFile{askinclude-a}
1233 \gdef\AskInc@Answer{askinclude-a}
1234 \@input{askinclude-a.aux}
1235 \AskInc@AddFile{askinclude-b}
1236 \@input{askinclude-b.aux}
1237 \AskInc@AddFile{askinclude-c}
1238 \gdef\AskInc@Answer{askinclude-a,askinclude-c}
1239 \@input{askinclude-c.aux}
1240 \@EndChecks
1241 \end{filecontents*}
1242 \begin{document}
1243 \CheckInclude{a}{true}
1244 \CheckInclude{b}{false}
1245 \CheckInclude{c}{true}
1246 </test15>

1247 <*test16>
1248 \Stack{?A}
1249 \CheckAnswer{askinclude-a,askinclude-b,askinclude-c}
1250 \begin{filecontents*}{\jobname.aux}
1251 \gdef\AskInc@Answer{askinclude-a}
1252 \gdef\AskInc@MatchMethod{}
1253 |AddList
1254 |AddFile
1255 \AskInc@AddFile{askinclude-a}
1256 \AskInc@AddFile{askinclude-b}
1257 \end{filecontents*}
1258 \begin{filecontents*}{\jobname-aux.aux}
1259 \relax|space
1260 \gdef\AskInc@Answer{}
1261 \gdef\AskInc@MatchMethod{}
1262 |AddList
1263 |AddFile
1264 \AskInc@AddFile{askinclude-a}
1265 \gdef\AskInc@Answer{askinclude-a}
1266 \@input{askinclude-a.aux}
1267 \AskInc@AddFile{askinclude-b}
1268 \gdef\AskInc@Answer{askinclude-a,askinclude-b}
1269 \@input{askinclude-b.aux}
1270 \AskInc@AddFile{askinclude-c}
1271 \gdef\AskInc@Answer{askinclude-a,askinclude-b,askinclude-c}
1272 \@input{askinclude-c.aux}
1273 \@EndChecks
1274 \end{filecontents*}
1275 \begin{document}
1276 \CheckInclude{a}{true}
1277 \CheckInclude{b}{true}
1278 \CheckInclude{c}{true}
1279 </test16>

```

```

1280 <*test17>
1281 \Stack{?N}
1282 \CheckAnswer{}
1283 \begin{filecontents*}{\jobname.aux}
1284 \gdef\AskInc@Answer{askinclude-a}
1285 \gdef\AskInc@MatchMethod{}
1286 |AddList
1287 |AddFile
1288 \AskInc@AddFile{askinclude-a}
1289 \AskInc@AddFile{askinclude-b}
1290 \end{filecontents*}
1291 \begin{filecontents*}{\jobname-aux.aux}
1292 \relax|space
1293 \gdef\AskInc@Answer{}
1294 \gdef\AskInc@MatchMethod{}
1295 |AddList
1296 |AddFile
1297 \AskInc@AddFile{askinclude-a}
1298 \@input{askinclude-a.aux}
1299 \AskInc@AddFile{askinclude-b}
1300 \@input{askinclude-b.aux}
1301 \AskInc@AddFile{askinclude-c}
1302 \@input{askinclude-c.aux}
1303 \@EndChecks
1304 \end{filecontents*}
1305 \begin{document}
1306 \CheckInclude{a}{false}
1307 \CheckInclude{b}{false}
1308 \CheckInclude{c}{false}
1309 </test17>

1310 <*test18>
1311 \Stack{}
1312 \CheckAnswer{askinclude-a}
1313 \begin{filecontents*}{\jobname.aux}
1314 \gdef\AskInc@Answer{askinclude-a}
1315 \gdef\AskInc@MatchMethod{}
1316 |AddList
1317 |AddFile
1318 \AskInc@AddFile{askinclude-a}
1319 \AskInc@AddFile{askinclude-b}
1320 \end{filecontents*}
1321 \begin{filecontents*}{\jobname-aux.aux}
1322 \relax|space
1323 \gdef\AskInc@Answer{askinclude-a}
1324 \gdef\AskInc@MatchMethod{}
1325 |AddList
1326 |AddFile
1327 \AskInc@AddFile{askinclude-a}
1328 \@input{askinclude-a.aux}
1329 \AskInc@AddFile{askinclude-b}
1330 \@input{askinclude-b.aux}
1331 \AskInc@AddFile{askinclude-c}
1332 \@input{askinclude-c.aux}
1333 \@EndChecks
1334 \end{filecontents*}
1335 \begin{document}
1336 \CheckInclude{a}{false}
1337 \CheckInclude{b}{true}
1338 \CheckInclude{c}{false}
1339 </test18>

1340 <*test19>
1341 \Stack{?N}

```

```

1342 \CheckAnswer{}
1343 \begin{filecontents*}{\jobname.aux}
1344 \gdef\AskInc@Answer{askinclude-a}
1345 \gdef\AskInc@MatchMethod{}
1346 |AddList
1347 |AddFile
1348 \AskInc@AddFile{askinclude-a}
1349 \AskInc@AddFile{askinclude-b}
1350 \end{filecontents*}
1351 \begin{filecontents*}{\jobname-aux.aux}
1352 \relax|space
1353 \gdef\AskInc@Answer{}
1354 \gdef\AskInc@MatchMethod{}
1355 |AddList
1356 |AddFile
1357 \AskInc@AddFile{askinclude-a}
1358 \@input{askinclude-a.aux}
1359 \AskInc@AddFile{askinclude-b}
1360 \@input{askinclude-b.aux}
1361 \AskInc@AddFile{askinclude-c}
1362 \@input{askinclude-c.aux}
1363 \@EndChecks
1364 \end{filecontents*}
1365 \ifx\AtBeginDocumentMode Y%
1366   \AtBeginDocument{\includeonly{askinclude-b}}%
1367 \else
1368   \includeonly{askinclude-b}%
1369 \fi
1370 \begin{document}
1371 \CheckInclude{a}{false}
1372 \CheckInclude{b}{false}
1373 \CheckInclude{c}{false}
1374 \end{test19}

```

### 3.6 Compatibility for changing from old to new

```

1375 \begin{test20}
1376 \Stack{}
1377 \CheckAnswer{*}
1378 \begin{filecontents*}{\jobname.aux}
1379 \includeonly{*}
1380 \end{filecontents*}
1381 \begin{filecontents*}{\jobname-aux.aux}
1382 \relax|space
1383 \gdef\AskInc@Answer{*}
1384 \gdef\AskInc@MatchMethod{}
1385 |AddList
1386 |AddFile
1387 \AskInc@AddFile{askinclude-a}
1388 \@input{askinclude-a.aux}
1389 \AskInc@AddFile{askinclude-b}
1390 \@input{askinclude-b.aux}
1391 \AskInc@AddFile{askinclude-c}
1392 \@input{askinclude-c.aux}
1393 \@EndChecks
1394 \end{filecontents*}
1395 \begin{document}
1396 \CheckInclude{a}{true}
1397 \CheckInclude{b}{true}
1398 \CheckInclude{c}{true}
1399 \end{test20}

1400 \begin{test21}
1401 \Stack{}

```



```

1402 \CheckAnswer{-}
1403 \begin{filecontents*}{\jobname.aux}
1404 \@includeonly{}
1405 \end{filecontents*}
1406 \begin{filecontents*}{\jobname-aux.aux}
1407 \relax|space
1408 \gdef\AskInc@Answer{-}
1409 \gdef\AskInc@MatchMethod{}
1410 |AddList
1411 |AddFile
1412 \AskInc@AddFile{askinclude-a}
1413 \@input{askinclude-a.aux}
1414 \AskInc@AddFile{askinclude-b}
1415 \@input{askinclude-b.aux}
1416 \AskInc@AddFile{askinclude-c}
1417 \@input{askinclude-c.aux}
1418 \@EndChecks
1419 \end{filecontents*}
1420 \begin{document}
1421 \CheckInclude{a}{false}
1422 \CheckInclude{b}{false}
1423 \CheckInclude{c}{false}
1424 </test21>

1425 <*test22>
1426 \Stack{{}}
1427 \CheckAnswer{askinclude-a,askinclude-c}
1428 \begin{filecontents*}{\jobname.aux}
1429 \@includeonly{askinclude-a,askinclude-c}
1430 \end{filecontents*}
1431 \begin{filecontents*}{\jobname-aux.aux}
1432 \relax|space
1433 \gdef\AskInc@Answer{askinclude-a,askinclude-c}
1434 \gdef\AskInc@MatchMethod{}
1435 |AddList
1436 |AddFile
1437 \AskInc@AddFile{askinclude-a}
1438 \@input{askinclude-a.aux}
1439 \AskInc@AddFile{askinclude-b}
1440 \@input{askinclude-b.aux}
1441 \AskInc@AddFile{askinclude-c}
1442 \@input{askinclude-c.aux}
1443 \@EndChecks
1444 \end{filecontents*}
1445 \begin{document}
1446 \CheckInclude{a}{true}
1447 \CheckInclude{b}{false}
1448 \CheckInclude{c}{true}
1449 </test22>

```

### 3.7 Regular expressions

```

1450 <*test23>
1451 \Stack{{makematch}{*,!askinclude-b}}
1452 \CheckAnswer{*,!askinclude-b}
1453 \CheckMatchMethod{makematch}
1454 \begin{filecontents*}{\jobname.aux}
1455 \relax
1456 \end{filecontents*}
1457 \begin{filecontents*}{\jobname-aux.aux}
1458 \relax|space
1459 \gdef\AskInc@Answer{*,!askinclude-b}
1460 \gdef\AskInc@MatchMethod{makematch}
1461 |AddList

```

```

1462 |AddFile
1463 \AskInc@AddFile{askinclude-a}
1464 |AddMatch
1465 \AskInc@AddMatch{askinclude-a}
1466 \@input{askinclude-a.aux}
1467 \AskInc@AddFile{askinclude-b}
1468 \@input{askinclude-b.aux}
1469 \AskInc@AddFile{askinclude-c}
1470 \AskInc@AddMatch{askinclude-c}
1471 \@input{askinclude-c.aux}
1472 \@EndChecks
1473 \end{filecontents*}
1474 \begin{document}
1475 \CheckInclude{a}{true}
1476 \CheckInclude{b}{false}
1477 \CheckInclude{c}{true}
1478 </test23>

1479 <*test24>
1480 \Stack{{pdfmatch}}{.,!askinclude-b}}
1481 \CheckAnswer{.,!askinclude-b}
1482 \CheckMatchMethod{pdfmatch}
1483 \begin{filecontents*}{\jobname.aux}
1484 \relax
1485 \end{filecontents*}
1486 \begin{filecontents*}{\jobname-aux.aux}
1487 \relax|space
1488 \gdef\AskInc@Answer{.,!askinclude-b}
1489 \gdef\AskInc@MatchMethod{pdfmatch}
1490 |AddList
1491 |AddFile
1492 \AskInc@AddFile{askinclude-a}
1493 |AddMatch
1494 \AskInc@AddMatch{askinclude-a}
1495 \@input{askinclude-a.aux}
1496 \AskInc@AddFile{askinclude-b}
1497 \@input{askinclude-b.aux}
1498 \AskInc@AddFile{askinclude-c}
1499 \AskInc@AddMatch{askinclude-c}
1500 \@input{askinclude-c.aux}
1501 \@EndChecks
1502 \end{filecontents*}
1503 \begin{document}
1504 \CheckInclude{a}{true}
1505 \CheckInclude{b}{false}
1506 \CheckInclude{c}{true}
1507 </test24>

```

### 3.8 Common end

```

1508 <*test0>
1509 \end{document}
1510 </test0>

```

## 4 Installation

### 4.1 Download

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

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

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

---

<sup>1</sup>[ftp://ftp.ctan.org/tex-archive/](http://ftp.ctan.org/tex-archive/)

**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.

## 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 askinclude.dtx
```

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

<code>askinclude.sty</code>	→ <code>tex/latex/oberdiek/askinclude.sty</code>
<code>askinclude.pdf</code>	→ <code>doc/latex/oberdiek/askinclude.pdf</code>
<code>test/askinclude-a.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-a.tex</code>
<code>test/askinclude-b.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-b.tex</code>
<code>test/askinclude-c.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-c.tex</code>
<code>test/askinclude-test1.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test1.tex</code>
<code>test/askinclude-test2.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test2.tex</code>
<code>test/askinclude-test3.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test3.tex</code>
<code>test/askinclude-test4.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test4.tex</code>
<code>test/askinclude-test5.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test5.tex</code>
<code>test/askinclude-test6.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test6.tex</code>
<code>test/askinclude-test7.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test7.tex</code>
<code>test/askinclude-test8.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test8.tex</code>
<code>test/askinclude-test9.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test9.tex</code>
<code>test/askinclude-test10.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test10.tex</code>
<code>test/askinclude-test11.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test11.tex</code>
<code>test/askinclude-test12.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test12.tex</code>
<code>test/askinclude-test13.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test13.tex</code>
<code>test/askinclude-test14.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test14.tex</code>
<code>test/askinclude-test15.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test15.tex</code>
<code>test/askinclude-test16.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test16.tex</code>
<code>test/askinclude-test17.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test17.tex</code>
<code>test/askinclude-test18.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test18.tex</code>
<code>test/askinclude-test19.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test19.tex</code>
<code>test/askinclude-test20.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test20.tex</code>
<code>test/askinclude-test21.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test21.tex</code>
<code>test/askinclude-test22.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test22.tex</code>
<code>test/askinclude-test23.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test23.tex</code>
<code>test/askinclude-test24.tex</code>	→ <code>doc/latex/oberdiek/test/askinclude-test24.tex</code>
<code>askinclude.dtx</code>	→ <code>source/latex/oberdiek/askinclude.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`.

## 4.4 Refresh file name databases

If your  $\text{\TeX}$  distribution (`te\TeX`, `mik\TeX`, ...) relies on file name databases, you must refresh these. For example, `te\TeX` 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 askinclude.pdf unpack_files output .
```

**Unpacking with  $\text{\LaTeX}$ .** The `.dtx` chooses its action depending on the format:

**plain- $\text{\TeX}$ :** Run `docstrip` and extract the files.

**$\text{\LaTeX}$ :** Generate the documentation.

If you insist on using  $\text{\LaTeX}$  for `docstrip` (really, `docstrip` does not need  $\text{\LaTeX}$ ), then inform the autodetect routine about your intention:

```
latex \let\install=y\input{askinclude.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\LaTeX`:

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

## 5 Acknowledgement

**Tom Scavo** suggested saving the include file list from the previous run and a simplified user interface.

**Joseph Wright** suggested questions for each `\include`.

## 6 References

- [1] Posting in mailing list `texlive@tug.org`  
From: Robin Fairbairns  
Subject: licence on includeonly.sty  
Date: Sat Oct 13 13:28:47 CEST 2007  
<http://www.tug.org/pipermail/tex-live/2007-October/014928.html>

- [2] Thread in newsgroup `comp.text.tex`  
 From: Herbert Voss  
 Subject: askinclude.sty  
 Date: Mon, 15 Oct 2007 10:06:45 +0200  
 Message-ID: fev72m\$b35\$02\$2@news.t-online.com  
[http://groups.google.com/group/comp.text.tex/browse\\_thread/thread/728a3e73c04c6789/349e31865c40d00c?#349e31865c40d00c](http://groups.google.com/group/comp.text.tex/browse_thread/thread/728a3e73c04c6789/349e31865c40d00c?#349e31865c40d00c)
- [3] David Kastrup: *makematch.sty: Pattern matching with wildcards*; 2007/02/23; [CTAN:macros/latex/contrib/qstest/makematch.dtx](http://CTAN:macros/latex/contrib/qstest/makematch.dtx).
- [4] Daniel H. Luecking, Donald Arsenau: *excludeonly.sty: eponymous command opposite to \includeonly*; 2003/03/14; [CTAN:macros/latex/contrib/misc/excludeonly.sty](http://CTAN:macros/latex/contrib/misc/excludeonly.sty).

## 7 History

### [1991/11/05 v0.95]

- First beta-test version (asked two questions).

### [1991/11/05 v0.99]

- Second beta-test version (asks one question; uses ‘\*’ and ‘-’).

### [1991/11/07 v1.0]

- Changed prompts (sent to `comp.text.tex`): “Interactive inclusion of files in LaTeX”<sup>2</sup>

### [1991/12/13 v1.1]

- *Unknown*.

### [1994/11/02 v1.2e]

- Two small changes to run under LaTeX2e [RF].
- Additional documentation [TRS].

### [2007/10/15 v1.0 (askinc)]

- Reimplementation, published as askinc in newsgroup `comp.text.tex`: “Re: askinclude.sty”<sup>3</sup>
- Listing of included files, collected from the previous run before the question.
- No error message by `.aux` file if package is removed.

### [2007/10/23 v2.0]

- The next version of askinc becomes version 2.0 of askinclude.
- ‘?’ added for questioning each file.
- `\includeonly` has precedence until package loading. Then it is disabled by a warning.
- Documentation rewritten/added.
- Tests added.

<sup>2</sup>Url: <http://groups.google.com/group/comp.text.tex/msg/34b770a7c277e79c>

<sup>3</sup>Url: <http://groups.google.com/group/comp.text.tex/msg/2d638b84a858fdc8>

- Code is not changed.
- URLs updated.

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.

A

1361, 1387, 1389, 1391, 1412, 1414, 1416, 1437, 1439, 1441, 1463, 1467, 1469, 1492, 1496, 1498	\AskInc@MatchFiles . . . . .
	. . . . . 146, 243, 271, 515, 555, 833
\AskInc@AddList . . . . .	\AskInc@MatchInit . . . . . 412, 420
. . . . . 60, 85, 515, 815, 828, 833	\AskInc@MatchList . . . . . 423,
\AskInc@AddMatch . . . . . 514, 521,	447, 457, 491, 494, 500, 501, 503
527, 832, 1465, 1470, 1494, 1499	\AskInc@MatchMethod 120, 130, 143,
\AskInc@Answer . 115, 126, 127, 128,	230, 232, 234, 265, 297, 300,
224, 226, 228, 259, 263, 270,	394, 400, 403, 406, 434, 788,
304, 383, 393, 398, 409, 422,	847, 872, 897, 922, 947, 972,
532, 574, 575, 577, 580, 778,	999, 1027, 1055, 1081, 1107,
846, 871, 896, 921, 946, 971,	1129, 1139, 1160, 1169, 1190,
976, 981, 998, 1003, 1006, 1009,	1199, 1220, 1229, 1252, 1261,
1026, 1031, 1034, 1037, 1054,	1285, 1294, 1315, 1324, 1345,
1059, 1080, 1085, 1106, 1111,	1354, 1384, 1409, 1434, 1460, 1489
1128, 1138, 1159, 1168, 1189,	\AskInc@MatchOption . . . . .
1198, 1219, 1228, 1233, 1238,	. 172, 176, 187, 234, 236, 239,
1251, 1260, 1265, 1268, 1271,	324, 366, 369, 373, 379, 406, 407
1284, 1293, 1314, 1323, 1344,	\AskInc@Next . . . . . 364,
1353, 1383, 1408, 1433, 1459, 1488	385, 389, 596, 599, 602, 626, 630
\AskInc@AskFile . . . . . 103, 107, 395	\AskInc@OrgInclude . . . . . 101, 109
\AskInc@AskQuestion . . . . . 572, 583	\AskInc@OrgIncludeOnly . . . . .
\AskInc@AskResult . . . . .	. . . . . 153, 396, 404, 408, 413
. 573, 592, 594, 595, 598, 601,	\AskInc@Pattern . . . 497, 498, 501, 504
604, 606, 609, 611, 614, 616, 620	\AskInc@PdfMatchAvailabletrue . . 169
\AskInc@AtBeginDocument . . . . . 45, 50	\AskInc@PdfMatchInit . . . . . 437, 452
\AskInc@AtEnd . . . . . 20, 21, 150, 633	\AskInc@PreviousAnswer . . . . .
\AskInc@CharAll . . . . . 197, 604	. . . . . 228, 535, 543, 547, 554
\AskInc@CharAsterisk . . . . .	\AskInc@PreviousMatches . . . . .
. . . . . 190, 224, 259, 399, 536, 543	. . . . . 254, 281, 282, 285
\AskInc@CharDefault . . . . . 199, 614	\AskInc@PreviousMatchMethod . . . . .
\AskInc@CharExclam . . . . . 194, 469	. . . . . 232, 538, 549
\AskInc@CharHash 57, 61, 63, 65, 72, 74	\AskInc@QuestionHeader 246, 337, 571
\AskInc@CharMinus . . . . .	\AskInc@SanitizeAnswer 200, 226, 362
. . . . . 191, 262, 263, 402, 547	\AskInc@SaveAnswer . . . . . 530, 579
\AskInc@CharNo . . . . .	\AskInc@StringMakeMatch . . . . .
. . . . . 196, 546, 562, 566, 601, 611	. . . . . 163, 172, 312, 368, 369, 434
\AskInc@CharNone . . . . . 198, 609	\AskInc@StringNoRegExp . . . . 162, 365
\AskInc@CharPlus . . 192, 260, 274, 280	\AskInc@StringPdfMatch . . . . .
\AskInc@CharQuestion . . . . . 193, 392	. . . . . 164, 176, 236, 316, 371, 373
\AskInc@CharYes . . . . .	\AskInc@Temp . . . . . 270,
. 195, 544, 558, 563, 573, 598, 606	271, 273, 444, 445, 447, 448,
\AskInc@CheckMatch . . . . . 457, 464	468, 469, 486, 488, 490, 498,
\AskInc@Default . . . . . 544, 546,	554, 555, 557, 564, 565, 584, 593
558, 562, 563, 566, 589, 595, 616	\AskInc@TimeoutFiles . . . . . 253, 338
\AskInc@DoAskFile . . . . . 395, 541	\AskInc@TimeoutMainQuestion 328, 341
\AskInc@DoInclude . . . . . 418	\AskInc@TimeoutMatchMethod . 307, 340
\AskInc@DoMatch . . . . . 425, 443, 454	\AskInc@TimeoutMethod . . . . .
\AskInc@File . . . . . 258, 273,	. . . . . 309, 312, 316, 322
279, 282, 286, 542, 557, 565, 588	\AskInc@TimeoutPreviousAnswer . .
\AskInc@Files 85, 100, 255, 258, 564, 828	. . . . . 294, 339
\AskInc@FilterPattern . . . . . 494, 496	\AskInc@WriteAddFile . . . . . 94, 106
\AskInc@Foundfalse . . . . . 455, 472	\AskInc@WriteAddMatch . . . . . 426, 524
\AskInc@Foundtrue . . . . . 479	\AskInc@WriteAux . . . . . 49, 59,
\AskInc@Include . . . . . 418	83, 96, 114, 119, 145, 513, 526, 531
\AskInc@InvalidPattern 474, 481, 485	\AskInc@WriteDefAddFile . . . . . 81, 95
\AskInc@MainAnswer . . . . . 385, 391	\AskInc@WriteDefAddList . 58, 82, 512
\AskInc@MainQuestion . . . . . 336, 632	\AskInc@WriteDefAddMatch . . . 511, 525
\AskInc@MakeMatchInit . . . . . 435, 441	\AtBeginDocument . . . . . 45, 48, 91,
\AskInc@Mark . . 260, 262, 274, 279, 280	112, 222, 521, 632, 703, 765, 1366
\AskInc@MatchFile . . . . 104, 108, 424	\AtBeginDocumentMode . . 651, 702, 1365
	\AtEndDocument . . . . . 746, 766
	\AtEndOfPackage . . . . . 111





<code>\includeonly</code> . . . . .	<code>\providecommand</code> . . . . . 48,
. . . 134, 154, 155, 648, 1366, 1368	60, 84, 100, 125, 514, 815, 827, 832
<code>\IncludeTests</code> . . . . . 716	<code>\ProvidesPackage</code> . . . . . 43
<b>J</b>	
<code>\jobname</code> . . . . . 799,	
800, 841, 844, 866, 869, 891,	
894, 916, 919, 941, 944, 966,	
969, 993, 996, 1021, 1024, 1049,	
1052, 1075, 1078, 1101, 1104,	
1127, 1136, 1158, 1166, 1188,	
1196, 1218, 1226, 1250, 1258,	
1283, 1291, 1313, 1321, 1343,	
1351, 1378, 1381, 1403, 1406,	
1428, 1431, 1454, 1457, 1483, 1486	
<b>L</b>	
<code>\LogTests</code> . . . . . 717	
<code>\loop</code> . . . . . 664, 674	
<b>M</b>	
<code>\makeatletter</code> . . . . . 661, 718	
<code>\makeatother</code> . . . . . 697	
<code>\MakeMatcher</code> . . . . . 445	
<code>\MatchMethodText</code> . . . . . 783, 789	
<code>\MessageBreak</code> . . . . . 134, 158, 181, 376, 620	
<b>N</b>	
<code>\NeedsTeXFormat</code> . . . . . 2	
<code>\newcommand</code> 101, 153, 743, 756, 772, 782	
<code>\newif</code> . . . . . 165, 453	
<b>O</b>	
<code>\org@filec@ntents</code> . . . . . 804, 807	
<code>\org@typein</code> . . . . . 719, 725	
<b>P</b>	
<code>\PackageWarning</code> . . . . . 156, 375	
<code>\PackageWarningNoLine</code> . . . . .	
. . . . . 133, 180, 487, 619	
<code>\pdffiledump</code> . . . . . 796	
<code>\pdffilesize</code> . . . . . 796	
<code>\pdfmatch</code> . . . . . 182, 317, 377, 470, 477, 488	
<code>\pdfunescapehex</code> . . . . . 795	
<code>\ProcessOptions</code> . . . . . 140, 189	
<b>R</b>	
<code>\RangeCatcodeInvalid</code> . . . . .	
. . . . . 672, 681, 682, 683, 684	
<code>\renewcommand</code> . . . . . 105, 155	
<code>\repeat</code> . . . . . 671, 678	
<code>\RequirePackage</code> . . . . . 421, 442	
<code>\RestoreCatcodes</code> . . . . . 662, 665, 666, 696	
<b>S</b>	
<code>\space</code> . . . . . 134,	
157, 218, 310, 317, 324, 330, 360	
<code>\Stack</code> . . . . . 743, 839, 864,	
889, 914, 939, 964, 991, 1019,	
1047, 1073, 1099, 1125, 1156,	
1186, 1216, 1248, 1281, 1311,	
1341, 1376, 1401, 1426, 1451, 1480	
<b>T</b>	
<code>\Test</code> . . . . . 680, 699	
<code>\the</code> . . . . . 8, 9, 10, 11, 22, 667	
<code>\TMP@EnsureCode</code> . . . . . 19,	
26, 27, 28, 29, 30, 31, 32, 33,	
34, 35, 36, 37, 38, 39, 40, 41, 42	
<code>\typein</code> . . . . . 359, 587, 719, 720	
<code>\typeout</code> . . . . . 247, 248, 249, 250,	
251, 257, 279, 291, 295, 304,	
305, 308, 320, 323, 329, 330,	
332, 333, 334, 361, 591, 737, 738	
<b>U</b>	
<code>\unexpanded</code> 728, 733, 778, 779, 788, 789	
<code>\usepackage</code> . . . . . 658, 659, 704, 708	
<b>W</b>	
<code>\write</code> . . . . . 52, 768	
<b>X</b>	
<code>\x</code> . . . . . 6, 14	
<b>Z</b>	
<code>\zap@space</code> . . . . . 217	