

The askinclude package

Pablo A. Straub

Heiko Oberdiek (maintainer)
<oberdiek@uni-freiburg.de>

2007/10/23 v2.0

Abstract

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

Contents

1	Documentation	2
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
2	Implementation	6
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	7
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	9
2.9	<code>\includeonly</code>	10
2.10	Options	10
2.11	Strings	11
2.12	Set defaults	11
2.13	Questions	12
2.13.1	Generic question header	12
2.13.2	Display files	12
2.13.3	Previous answer	13
2.13.4	Current regular expression support	13
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

3	Test	20
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
4	Installation	34
4.1	Download	34
4.2	Bundle installation	34
4.3	Package installation	35
4.4	Refresh file name databases	35
4.5	Some details for the interested	35
5	Acknowledgement	36
6	References	36
7	History	37
	[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
8	Index	37

1 Documentation

1.1 User interface

The `\include` feature of \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^AT_EX 2.09: The package works as `\documentstyle` option in compatibility mode of L^AT_EX 2_ε. It is possible that it will not work in pure L^AT_EX 2.09.

2 Implementation

2.1 Header and catcodes

```

1 <*package>
2 \NeedsTeXFormat{LaTeX2e}
3 \expandafter\edef\csname AskInc@AtEnd\endcsname{%
4   \catcode64 \the\catcode64\relax
5 }
6 \catcode64 11 % @

```

```

7 \def\TMP@EnsureCode#1#2{%
8   \edef\AskInc@AtEnd{%
9     \AskInc@AtEnd
10    \catcode#1 \the\catcode#1\relax
11  }%
12  \catcode#1 #2\relax
13 }
14 \TMP@EnsureCode{33}{12}% !
15 \TMP@EnsureCode{39}{12}% '
16 \TMP@EnsureCode{40}{12}% (
17 \TMP@EnsureCode{41}{12}% )
18 \TMP@EnsureCode{42}{12}% *
19 \TMP@EnsureCode{43}{12}% +
20 \TMP@EnsureCode{44}{12}% ,
21 \TMP@EnsureCode{45}{12}% -
22 \TMP@EnsureCode{46}{12}% .
23 \TMP@EnsureCode{47}{12}% /
24 \TMP@EnsureCode{58}{12}% :
25 \TMP@EnsureCode{61}{12}% =
26 \TMP@EnsureCode{62}{12}% >
27 \TMP@EnsureCode{63}{12}% ?
28 \TMP@EnsureCode{91}{12}% [
29 \TMP@EnsureCode{93}{12}% ]
30 \TMP@EnsureCode{96}{12}% '
31 \ProvidesPackage{askinclude}%
32 [2007/10/23 v2.0 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.

```

33 \AtBeginDocument{%
34   \let\AskInc@AtBeginDocument\@firstofone
35 }
36 \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.

```

37 \def\AskInc@WriteAux#1#2{%
38   \AskInc@AtBeginDocument{%
39     \if@files
40       \immediate\write\@mainaux{#1}%
41       #2%
42     \fi
43   }%
44 }

```

2.4 Remember \include files

`\AskInc@CharHash`

```

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

```

`\AskInc@WriteDefAddList`

```

46 \def\AskInc@WriteDefAddList{%
47   \AskInc@WriteAux{%

```

\AskInc@AddList

```
48 \string\providecommand*\string\AskInc@AddList[2]{%
49 \string\ifcase\string\ifx\AskInc@CharHash1\string\@undefined
50 \string\else
51 \string\ifx\AskInc@CharHash1\string\relax
52 \string\else
53 \string\ifx\AskInc@CharHash1\string\@empty
54 \string\else
55 1%
56 \string\fi
57 \string\fi
58 \string\fi
59 0 %
60 \string\gdef\AskInc@CharHash1{\AskInc@CharHash2}%
61 \string\else
62 \string\g@addto@macro\AskInc@CharHash1{\AskInc@CharHash2}%
63 \string\fi
64 }%

65 }{%
66 \global\let\AskInc@WriteDefAddList\relax
67 }%
68 }
```

\AskInc@WriteDefAddFile

```
69 \def\AskInc@WriteDefAddFile{%
70 \AskInc@WriteDefAddList
71 \AskInc@WriteAux{%
```

\AskInc@AddFile

```
72 \string\providecommand\string\AskInc@AddFile{%
73 \string\AskInc@AddList\string\AskInc@Files
74 }%

75 }{%
76 \global\let\AskInc@WriteDefAddFile\relax
77 }%
78 }
```

\AskInc@AddFile

```
79 \AtBeginDocument{%
80 \global\let\AskInc@AddFile\@gobble
81 }
```

\AskInc@WriteAddFile

```
82 \def\AskInc@WriteAddFile#1{%
83 \AskInc@WriteDefAddFile
84 \AskInc@WriteAux{%
85 \string\AskInc@AddFile{#1}%
86 }{%}%
87 }
```

\AskInc@Files

```
88 \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
89 \newcommand{\AskInc@OrgInclude}{}
90 \let\AskInc@OrgInclude\include

\AskInc@AskFile
91 \let\AskInc@AskFile\@gobble

\AskInc@MatchFile
92 \let\AskInc@MatchFile\@gobble

\include
93 \renewcommand*{\include}[1]{%
94   \AskInc@WriteAddFile{#1}%
95   \AskInc@AskFile{#1}%
96   \AskInc@MatchFile{#1}%
97   \AskInc@OrgInclude{#1}%
98 }

```

2.6 Store answer and match method in .aux file

```

99 \AtEndOfPackage{%
100   \AtBeginDocument{%
101     \ifundefined{AskInc@Answer}{}{}%
102     \AskInc@WriteAux{%
103       \string\gdef\string\AskInc@Answer{\AskInc@Answer}%
104     }{}%
105   }%
106   \ifundefined{AskInc@MatchMethod}{}{}%
107   \AskInc@WriteAux{%
108     \string\gdef\string\AskInc@MatchMethod{\AskInc@MatchMethod}%
109   }{}%
110 }%
111 }%
112 }

```

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`.

```

113 \providecommand*{\@includeonly}[1]{%
114   \gdef\AskInc@Answer{#1}%
115   \ifx\AskInc@Answer\@empty
116     \gdef\AskInc@Answer{-}%
117   \fi
118   \global\let\AskInc@MatchMethod\@empty
119 }

```

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.

```

120 \if@partsw
121   \PackageWarningNoLine{askinclude}{%
122     Previous \string\includeonly\space detected,\MessageBreak
123     therefore asking for files is suppressed%
124   }%

```

Dummy options.

```

125 \DeclareOption{makematch}{}%
126 \DeclareOption{pdfmatch}{}%

```

```

127 \DeclareOption{noregexp}{}%
128 \ProcessOptions*%
Preserve \AskInc@MatchFiles.
129 \@ifundefined{AskInc@MatchFiles}{%
130 }{%
131 \ifx\AskInc@MatchMethod\empty
132 \else
133 \AskInc@WriteAux{%
134 \string\gdef\string\AskInc@MatchFiles{\AskInc@MatchFiles}%
135 }{%
136 \fi
137 }%
138 \AskInc@AtEnd
139 \expandafter\endinput
140 \fi

```

2.9 \includeonly

\AskInc@OrgIncludeOnly

```

141 \newcommand{\AskInc@OrgIncludeOnly}{}%
142 \let\AskInc@OrgIncludeOnly\includeonly

```

\includeonly

```

143 \renewcommand*{\includeonly}[1]{%
144 \PackageWarning{askinclude}{%
145 \string\includeonly\space is managed by package ‘askinclude’,%
146 \MessageBreak
147 therefore it is ignored%
148 }%
149 }

```

2.10 Options

\AskInc@StringNoRegExp

```

150 \def\AskInc@StringNoRegExp{noregexp}

```

\AskInc@StringMakeMatch

```

151 \def\AskInc@StringMakeMatch{makematch}

```

\AskInc@StringPdfMatch

```

152 \def\AskInc@StringPdfMatch{pdfmatch}

```

\ifAskInc@PdfMatchAvailable

```

153 \newif\ifAskInc@PdfMatchAvailable
154 \begingroup\expandafter\expandafter\expandafter\endgroup
155 \expandafter\ifx\csname pdfmatch\endcsname\relax
156 \else
157 \AskInc@PdfMatchAvailabletrue
158 \fi

159 \DeclareOption{makematch}{%
160 \let\AskInc@MatchOption\AskInc@StringMakeMatch
161 }
162 \ifAskInc@PdfMatchAvailable
163 \DeclareOption{pdfmatch}{%
164 \let\AskInc@MatchOption\AskInc@StringPdfMatch
165 }%
166 \else
167 \DeclareOption{pdfmatch}{%
168 \PackageWarningNoLine{askinclude}{%

```

```

169      Option 'pdfmatch' is not available because\MessageBreak
170      your pdfTeX does not provide \string\pdfmatch
171    }%
172  }%
173 \fi
174 \DeclareOption{noregexp}{%
175   \let\AskInc@MatchOption\@empty
176 }
177 \ProcessOptions*

```

2.11 Strings

\AskInc@CharAsterisk

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

\AskInc@CharMinus

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

\AskInc@CharPlus

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

\AskInc@CharQuestion

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

\AskInc@CharExclam

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

\AskInc@CharYes

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

\AskInc@CharNo

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

\AskInc@CharAll

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

\AskInc@CharNone

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

\AskInc@CharDefault

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

2.12 Set defaults

\AskInc@SanitizeAnswer

```

188 \def\AskInc@SanitizeAnswer#1#2{%
189   \begingroup
190     \escapechar='\'%
191     \edef\[\{\string\[%
192     \edef\]\{\string\}%
193     \edef\(\{\string\(%
194     \edef\)\{\string\)%
195     \edef\*\{\string\}%
196     \edef\+\{\string\+}%
197     \edef\?\{\string\?}%
198     \edef\|\{\string\|}%
199     \edef\^\{\string\^}%
200     \edef\$\{\string\$}%
201     \edef\{\{\string\}%

```

```

202 \edef\}\{\string\}%
203 \edef\.\{\string\}%
204 \xdef#1{%
205 \expandafter\expandafter\expandafter\zap@space
206 \expandafter#2\space\@empty
207 }%
208 \endgroup
209 }

210 \AtBeginDocument{%
211 \@ifundefined{AskInc@Answer}{%
212 \global\let\AskInc@Answer\AskInc@CharAsterisk
213 }{%
214 \AskInc@SanitizeAnswer\AskInc@Answer\AskInc@Answer
215 }%
216 \global\let\AskInc@PreviousAnswer\AskInc@Answer
217 \@ifundefined{AskInc@MatchMethod}{%
218 \global\let\AskInc@MatchMethod\@empty
219 }{%
220 \global\let\AskInc@PreviousMatchMethod\AskInc@MatchMethod
221 \@ifundefined{AskInc@MatchOption}{%
222 \let\AskInc@MatchOption\AskInc@MatchMethod
223 }{%
224 \ifx\AskInc@MatchOption\AskInc@StringPdfMatch
225 \ifAskInc@PdfMatchAvailable
226 \else
227 \let\AskInc@MatchOption\@empty
228 \fi
229 \fi
230 \@ifundefined{AskInc@MatchFiles}{%
231 \let\AskInc@MatchFiles\@empty
232 }{%
233 }

```

2.13 Questions

2.13.1 Generic question header

\AskInc@QuestionHeader

```

234 \def\AskInc@QuestionHeader{%
235 \typeout{%
236 \typeout{*****}%
237 \typeout{*** Package askinclude Question ***}%
238 \typeout{*****}%
239 \typeout}%
240 }

```

2.13.2 Display files

\AskInc@TypeoutFiles

```

241 \def\AskInc@TypeoutFiles{%
242 \global\let\AskInc@PreviousMatches\@empty
243 \ifx\AskInc@Files\@empty
244 \else
245 \typeout{Files, found by previous run in \string\include:}%
246 \@for\AskInc@File:=\AskInc@Files\do{%
247 \ifx\AskInc@Answer\AskInc@CharAsterisk
248 \let\AskInc@Mark\AskInc@CharPlus
249 \else
250 \let\AskInc@Mark\AskInc@CharMinus
251 \ifx\AskInc@Answer\AskInc@CharMinus
252 \else

```

```

253         \ifx\AskInc@MatchMethod\@empty
254         \expandafter\@firstoftwo
255     \else
256         \expandafter\@secondoftwo
257     \fi
258     {\@for\AskInc@Temp:=\AskInc@Answer}%
259     {\@for\AskInc@Temp:=\AskInc@MatchFiles}%
260     \do{%
261         \ifx\AskInc@File\AskInc@Temp
262         \let\AskInc@Mark\AskInc@CharPlus
263         \fi
264     }%
265     \fi
266 \fi
267 \typeout{ \@spaces(\AskInc@Mark) \AskInc@File}%
268 \ifx\AskInc@Mark\AskInc@CharPlus
269     \ifx\AskInc@PreviousMatches\@empty
270     \global\let\AskInc@PreviousMatches\AskInc@File
271 \else
272     \expandafter\g@addto@macro
273     \expandafter\AskInc@PreviousMatches{%
274         \expandafter,\AskInc@File
275     }%
276 \fi
277 \fi
278 }%
279 \typeout{}%
280 \fi
281 }

```

2.13.3 Previous answer

\AskInc@TypeoutPreviousAnswer

```

282 \def\AskInc@TypeoutPreviousAnswer{%
283     \typeout{%
284         Previous answer (
285         \ifx\AskInc@MatchMethod\@empty
286             noregexp%
287         \else
288             \AskInc@MatchMethod
289         \fi
290         ):%
291     }%
292     \typeout{ \@spaces[\AskInc@Answer]]}%
293     \typeout{}%
294 }

```

2.13.4 Current regular expression support

\AskInc@TypeoutMatchMethod

```

295 \def\AskInc@TypeoutMatchMethod{%
296     \typeout{Regular expressions:}%
297     \AskInc@TypeoutMethod\@empty{noregexp}{%
298         \space disabled%
299     }%
300     \AskInc@TypeoutMethod\AskInc@StringMakeMatch{makematch}{%
301         enabled, using package ‘makematch’%
302     }%
303     \ifAskInc@PdfMatchAvailable
304     \AskInc@TypeoutMethod\AskInc@StringPdfMatch{pdfmatch}{%
305         \space enabled, using pdfTeX’s \string\pdfmatch
306     }%

```

```

307 \fi
308 \typeout{}%
309 }

```

\AskInc@TypeoutMethod

```

310 \def\AskInc@TypeoutMethod#1#2#3{%
311 \typeout{%
312 \ifx#1\AskInc@MatchOption \space--> \else\@spaces\space\fi
313 [#2] #3% hash-ok
314 }%
315 }

```

2.13.5 Main question text

\AskInc@TypeoutMainQuestion

```

316 \def\AskInc@TypeoutMainQuestion{%
317 \typeout{Which files do you want to include?}%
318 \typeout{ \@spaces[foo,bar] \space\space
319 comma separated file or pattern list}%
320 \typeout{ \@spaces[*] \@spaces\@spaces all files}%
321 \typeout{ \@spaces[-] \@spaces\@spaces no files}%
322 \typeout{ \@spaces[?] \@spaces\@spaces ask for each file}%
323 }

```

2.13.6 Main question

\AskInc@MainQuestion

```

324 \def\AskInc@MainQuestion{%
325 \AskInc@QuestionHeader
326 \AskInc@TypeoutFiles
327 \AskInc@TypeoutPreviousAnswer
328 \AskInc@TypeoutMatchMethod
329 \AskInc@TypeoutMainQuestion
330 \begingroup
331 \@makeother\*%
332 \@makeother\-%
333 \@makeother\?%
334 \@makeother\,%
335 \@makeother\!%
336 \@makeother\^%
337 \@makeother\$%
338 \@makeother\(%
339 \@makeother\)%
340 \@makeother\+%
341 \@makeother\[%
342 \@makeother\]%
343 \@makeother\\%
344 \@makeother\|%
345 \@makeother\:%
346 \@makeother\.%
347 \typein[\answer]%
348 { \@spaces[] \space\@spaces\@spaces use the previous answer}%
349 \typeout{}%
350 \AskInc@SanitizeAnswer\AskInc@@Answer\answer
351 \endgroup
352 \let\AskInc@Next\AskInc@MainQuestion
353 \ifx\AskInc@@Answer\AskInc@StringNoRegExp
354 \let\AskInc@MatchOption\@empty
355 \else
356 \ifx\AskInc@@Answer\AskInc@StringMakeMatch
357 \let\AskInc@MatchOption\AskInc@StringMakeMatch
358 \else

```

```

359     \ifx\AskInc@@Answer\AskInc@StringPdfMatch
360     \ifAskInc@PdfMatchAvailable
361     \let\AskInc@MatchOption\AskInc@StringPdfMatch
362     \else
363     \PackageWarning{askinclude}{%
364     Method 'pdfmatch' is not available because\MessageBreak
365     your pdfTeX does not provide \string\pdfmatch
366     }%
367     \let\AskInc@MatchOption\@empty
368     \fi
369     \else
370     \ifx\AskInc@@Answer\@empty
371     \global\let\AskInc@@Answer\AskInc@Answer
372     \fi
373     \let\AskInc@Next\AskInc@MainAnswer
374     \fi
375     \fi
376     \fi
377     \AskInc@Next
378 }

```

2.14 Evaluate answer of main question

\AskInc@MainAnswer

```

379 \def\AskInc@MainAnswer{%
380   \ifx\AskInc@@Answer\AskInc@CharQuestion
381     \global\let\AskInc@Answer\@empty
382     \global\let\AskInc@MatchMethod\@empty
383     \global\let\AskInc@AskFile\AskInc@DoAskFile
384     \AskInc@OrgIncludeOnly{%
385   \else
386     \global\let\AskInc@Answer\AskInc@@Answer
387     \ifx\AskInc@@Answer\AskInc@CharAsterisk
388       \global\let\AskInc@MatchMethod\@empty
389     \else
390       \ifx\AskInc@@Answer\AskInc@CharMinus
391         \global\let\AskInc@MatchMethod\@empty
392         \AskInc@OrgIncludeOnly{%
393       \else
394         \global\let\AskInc@MatchMethod\AskInc@MatchOption
395         \ifx\AskInc@MatchOption\@empty
396           \expandafter\AskInc@OrgIncludeOnly\expandafter{%
397             \AskInc@Answer
398           }%
399       \else
400         \AskInc@MatchInit
401         \AskInc@OrgIncludeOnly{%
402       \fi
403     \fi
404   \fi
405 \fi
406 \let\AskInc@Include\AskInc@DoInclude
407 }

```

\AskInc@MatchInit

```

408 \def\AskInc@MatchInit{%
409   \RequirePackage{kvsetkeys}[2007/09/29]%
410   \expandafter\comma@normalize\expandafter{\AskInc@Answer}%
411   \global\let\AskInc@MatchList\comma@list

```

\AskInc@MatchFile

```

412 \def\AskInc@MatchFile##1{%

```

```

413     \AskInc@DoMatch{##1}{%
414         \AskInc@WriteAddMatch{##1}%
415         \ifx\@partlist\@empty
416             \gdef\@partlist{##1}%
417         \else
418             \g@addto@macro\@partlist{,##1}%
419         \fi
420     }%
421 }%

422 \ifx\AskInc@MatchMethod\AskInc@StringMakeMatch
423     \AskInc@MakeMatchInit
424 \else
425     \AskInc@PdfMatchInit
426 \fi
427 \let\AskInc@MatchInit\@undefined
428 }

\AskInc@MakeMatchInit

429 \def\AskInc@MakeMatchInit{%
430     \RequirePackage{makematch}%

\AskInc@DoMatch

431 \def\AskInc@DoMatch##1{%
432     \def\AskInc@Temp{%
433         \MakeMatcher[,]\AskInc@Temp
434     }%
435     \expandafter\AskInc@Temp\expandafter{\AskInc@MatchList}%
436     \AskInc@Temp{##1}\@firstofone\@gobble
437 }%

438 \let\AskInc@MakeMatchInit\@undefined
439 }

\AskInc@PdfMatchInit

440 \def\AskInc@PdfMatchInit{%

\ifAskInc@Found

441     \newif\ifAskInc@Found

\AskInc@DoMatch

442 \def\AskInc@DoMatch##1{%
443     \AskInc@Foundfalse
444     \expandafter\comma@parse@normalized\expandafter
445     {\AskInc@MatchList}{\AskInc@CheckMatch{##1}}%
446     \ifAskInc@Found
447         \expandafter\@firstofone
448     \else
449         \expandafter\@gobble
450     \fi
451 }%

\AskInc@CheckMatch

452 \def\AskInc@CheckMatch##1##2{%
453     \AskInc@@CheckMatch##2\@empty\@nil{##1}%
454 }%

\AskInc@@CheckMatch

455 \def\AskInc@@CheckMatch##1##2\@nil##3{%
456     \def\AskInc@Temp{##1}%
457     \ifx\AskInc@Temp\AskInc@CharExclam

```



```

458     \ifcase\pdfmatch{##2}-{##3}=1 %
459     \or
460     \AskInc@Foundfalse
461     \else
462     \AskInc@InvalidPattern{##2}-{##1##2}%
463     \fi
464     \else
465     \ifcase\pdfmatch{##1##2}-{##3} %
466     \or
467     \AskInc@Foundtrue
468     \else
469     \AskInc@InvalidPattern{##1##2}-{##1##2}%
470     \fi
471     \fi
472 }%

```

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

```

473 \def\AskInc@InvalidPattern##1##2{%
474   \edef\AskInc@Temp{##1}%
475   \PackageWarningNoLine{askinclude}{%
476     Invalid pattern '\AskInc@Temp' for \string\pdfmatch
477   }%
478   \edef\AskInc@Temp{##2}%
479   \global\expandafter\let\expandafter\AskInc@MatchList
480   \expandafter\@empty
481   \expandafter\comma@parse@normalized\expandafter
482   {\AskInc@MatchList}\AskInc@FilterPattern
483 }%

```

\AskInc@FilterPattern

```

484 \def\AskInc@FilterPattern##1{%
485   \edef\AskInc@Pattern{##1}%
486   \ifx\AskInc@Temp\AskInc@Pattern
487   \else
488     \ifx\AskInc@MatchList\@empty
489     \global\let\AskInc@MatchList\AskInc@Pattern
490     \else
491     \expandafter\g@addto@macro\expandafter\AskInc@MatchList{%
492       \expandafter,\AskInc@Pattern
493     }%
494     \fi
495   \fi
496 }%

497 \let\AskInc@PdfMatchInit\@undefined
498 }

```

\AskInc@WriteDefAddMatch

```

499 \def\AskInc@WriteDefAddMatch{%
500   \AskInc@WriteDefAddList
501   \AskInc@WriteAux{%

```

\AskInc@AddMatch

```

502     \string\providecommand\string\AskInc@AddMatch{%
503     \string\AskInc@AddList\string\AskInc@MatchFiles
504     }%

505 }{%
506   \global\let\AskInc@WriteDefAddMatch\relax
507 }%
508 }

```

\AskInc@AddMatch

```
509 \AtBeginDocument{%
510   \global\let\AskInc@AddMatch\@gobble
511 }
```

\AskInc@WriteAddMatch

```
512 \def\AskInc@WriteAddMatch#1{%
513   \AskInc@WriteDefAddMatch
514   \AskInc@WriteAux{%
515     \string\AskInc@AddMatch{#1}%
516   }{%}%
517 }
```

\AskInc@SaveAnswer

```
518 \def\AskInc@SaveAnswer{%
519   \AskInc@WriteAux{%
520     \string\gdef\string\AskInc@Answer{\AskInc@Answer}%
521   }{%}%
522 }
```

2.15 Ask for each file

\AskInc@PreviousAnswer

```
523 \@ifundefined{AskInc@PreviousAnswer}{%
524   \let\AskInc@PreviousAnswer\AskInc@CharAsterisk
525 }{}
```

\AskInc@PreviousMatchMethod

```
526 \@ifundefined{AskInc@PreviousMatchMethod}{%
527   \let\AskInc@PreviousMatchMethod\@empty
528 }{}
```

\AskInc@DoAskFile

```
529 \def\AskInc@DoAskFile#1{%
530   \edef\AskInc@File{#1}%
531   \ifx\AskInc@PreviousAnswer\AskInc@CharAsterisk
532     \let\AskInc@Default\AskInc@CharYes
533   \else
534     \let\AskInc@Default\AskInc@CharNo
535     \ifx\AskInc@PreviousAnswer\AskInc@CharMinus
536       \else
537         \ifx\AskInc@PreviousMatchMethod\@empty
538           \expandafter\@firstoftwo
539         \else
540           \expandafter\@secondoftwo
541         \fi
542         {\@for\AskInc@Temp:=\AskInc@PreviousAnswer}%
543         {\@for\AskInc@Temp:=\AskInc@MatchFiles}%
544         \do{%
545           \ifx\AskInc@Temp\AskInc@File
546             \let\AskInc@Default\AskInc@CharYes
547           \fi
548         }%
549       \fi
550     \ifx\AskInc@Default\AskInc@CharNo
551       \let\AskInc@Default\AskInc@CharYes
552       \@for\AskInc@Temp:=\AskInc@Files\do{%
553         \ifx\AskInc@Temp\AskInc@File
554           \let\AskInc@Default\AskInc@CharNo
555         \fi
556       }%
```

```

557 \fi
558 \fi
559 \let\AskInc@@QuestionHeader\AskInc@QuestionHeader
560 \AskInc@AskQuestion
561 \ifx\AskInc@AskResult\AskInc@CharYes
562 \ifx\AskInc@Answer@empty
563 \gdef\AskInc@Answer{#1}%
564 \else
565 \g@addto@macro\AskInc@Answer{, #1}%
566 \fi
567 \AskInc@SaveAnswer
568 \let\@partlist\AskInc@Answer
569 \fi
570 }

```

\AskInc@AskQuestion

```

571 \def\AskInc@AskQuestion{%
572 \let\AskInc@Temp\answer
573 \AskInc@@QuestionHeader
574 \let\AskInc@@QuestionHeader\relax
575 \typein[\answer]{%
576 \include '\AskInc@File'? [y]es, [n]o, [A]ll, [N]one, [D]efault %
577 (\AskInc@Default):%
578 }%
579 \typeout{}%
580 \let\AskInc@AskResult\answer
581 \let\answer\AskInc@Temp
582 \ifx\AskInc@AskResult@empty
583 \let\AskInc@AskResult\AskInc@Default
584 \let\AskInc@Next\relax
585 \else
586 \ifx\AskInc@AskResult\AskInc@CharYes
587 \let\AskInc@Next\relax
588 \else
589 \ifx\AskInc@AskResult\AskInc@CharNo
590 \let\AskInc@Next\relax
591 \else
592 \ifx\AskInc@AskResult\AskInc@CharAll
593 \gdef\AskInc@AskQuestion{%
594 \let\AskInc@AskResult\AskInc@CharYes
595 }%
596 \else
597 \ifx\AskInc@AskResult\AskInc@CharNone
598 \gdef\AskInc@AskQuestion{%
599 \let\AskInc@AskResult\AskInc@CharNo
600 }%
601 \else
602 \ifx\AskInc@AskResult\AskInc@CharDefault
603 \gdef\AskInc@AskQuestion{%
604 \let\AskInc@AskResult\AskInc@Default
605 }%
606 \else
607 \PackageWarningNoLine{askinclude}{%
608 Invalid answer [\AskInc@AskResult].\MessageBreak
609 Answer must be one letter out of [ynAND]%
610 }%
611 \fi
612 \fi
613 \fi
614 \let\AskInc@Next\AskInc@AskQuestion
615 \fi
616 \fi
617 \fi

```

```

618 \AskInc@Next
619 }

620 \AtBeginDocument{\AskInc@MainQuestion}

621 \AskInc@AtEnd
622 \end{package}

```

3 Test

3.1 Included files

```

623 \let\filea=Y
624 \chapter{Chapter A}
625 \let\filea=Y
626 \end{a}

627 \let\fileb=Y
628 \chapter{Chapter B}
629 \let\fileb=Y
630 \end{b}

631 \let\filec=Y
632 \chapter{Chapter C}
633 \let\filec=Y
634 \end{c}

```

3.2 Special test setup

```

635 \test18
636 \includeonly{askinclude-b}
637 \end{test18}

638 \test20 | test21 | test22
639 \let\AtBeginDocumentMode=N%
640 \test20 | test21 | test22

```

3.3 Common start

```

641 \test0
642 \ifx\CompatibilityMode Y%
643 \documentstyle[filecontents,qstest,askinclude]{book}%
644 \else
645 \documentclass{book}%
646 \usepackage{filecontents}%
647 \usepackage{qstest}%
648 \test1
649 \makeatletter
650 \def\RestoreCatcodes{%
651 \count@=0 %
652 \loop
653 \edef\RestoreCatcodes{%
654 \RestoreCatcodes
655 \catcode\the\count@=\the\catcode\count@\relax
656 }%
657 \ifnum\count@<255 %
658 \advance\count@\@ne
659 \repeat
660 \def\RangeCatcodeInvalid#1#2{%
661 \count@=#1\relax
662 \loop
663 \catcode\count@=#2 %
664 \ifnum\count@<#2\relax
665 \advance\count@\@ne
666 \repeat
667 }%

```

```

668 \def\Test#1\EndTest{%
669   \RangeCatcodeInvalid{0}{47}%
670   \RangeCatcodeInvalid{58}{64}%
671   \RangeCatcodeInvalid{91}{96}%
672   \RangeCatcodeInvalid{123}{255}%
673   \catcode'\@=12 %
674   \catcode'\|=0 %
675   \catcode'\{=1 %
676   \catcode'\}=2 %
677   \catcode'\#=6 %
678   \catcode'\[=12 %
679   \catcode'\]=12 %
680   \catcode'\%=14 %
681   \catcode'\ =10 %
682   \catcode13=5 %
683   #1\relax
684   \RestoreCatcodes
685   \makeatother
686 }%
687 \Test
688 </test1>
689 <!*test1>
690 \ifx\AtBeginDocumentMode Y%
691   \AtBeginDocument{%
692     \usepackage{askinclude}[2007/10/23]%
693   }%
694 \else
695 </!test1>
696   \usepackage{askinclude}[2007/10/23]%
697 <!*test1>
698 \fi
699 </!test1>
700 <*test1>
701 \EndTest
702 </test1>
703 \fi
704 \IncludeTests{*}
705 \LogTests{log}{*}{*}
706 \makeatletter
707 \let\org@typein\typein
708 \def\typein[#1]#2{%
709   \ifx\@Stack\@StackEmpty
710     \begin{qstest}{StackNonEmpty}{StackNonEmpty}%
711       \Expect{Error: empty stack}{}%
712     \end{qstest}%
713     \org@typein[#1]#2}%
714 \else
715   \edef#1{%
716     \unexpanded\expandafter\expandafter\expandafter{%
717       \expandafter\@car\@Stack\@nil
718     }%
719   }%
720   \xdef\@Stack{%
721     \unexpanded\expandafter\expandafter\expandafter{%
722       \expandafter\@cdr\@Stack\@nil
723     }%
724   }%
725   \typeout{%
726     \typeout{\string#1=#1}%
727   \fi
728 }
729 \def\@StackEmpty{\BOTTOM}

```

```

730 \let\@Stack\@StackEmpty
731 \newcommand*{\Stack}[1]{%
732   \def\@Stack{#1\BOTTOM}%
733 }
734 \AtEndDocument{%
735   \begin{qstest}{\StackUsed}{\StackUsed}%
736     \Expect*{%
737       \ifx\@Stack\@StackEmpty
738       \else
739         Error: Unused stack entries%
740       \fi
741     }{%}%
742   \end{qstest}%
743 }
744 \newcommand*{\CheckInclude}[2]{%
745   \include{askinclude-#1}%
746   \begin{qstest}{\CheckInclude-#1}{\CheckInclude-#1}%
747     \Expect*{%
748       \expandafter\ifx\csname file#1\endcsname Ytrue\else false\fi
749     }{#2}%
750   \end{qstest}%
751 }
752 \def\@EndChecks{}
753 \AtBeginDocument{%
754   \AtEndDocument{%
755     \if@filesw
756       \immediate\write\@mainaux{\string\@EndChecks}%
757     \fi
758   }%
759 }
760 \newcommand*{\CheckAnswer}[1]{%
761   \gdef\AnswerText{#1}%
762   \g@addto@macro\@EndChecks{\@CheckAnswer}%
763 }
764 \def\@CheckAnswer{%
765   \begin{qstest}{\CheckAnswer}{\CheckAnswer}%
766     \Expect*{\unexpanded\expandafter{\AskInc@Answer}}{%
767       *{\unexpanded\expandafter{\AnswerText}}%
768     }%
769 }
770 \newcommand*{\CheckMatchMethod}[1]{%
771   \gdef\MatchMethodText{#1}%
772   \g@addto@macro\@EndChecks{\@CheckMatchMethod}%
773 }
774 \def\@CheckMatchMethod{%
775   \begin{qstest}{\CheckMatchMethod}{\CheckMatchMethod}%
776     \Expect*{\unexpanded\expandafter{\AskInc@MatchMethod}}{%
777       *{\unexpanded\expandafter{\MatchMethodText}}%
778     }%
779 }
780 \def\@CheckAux{%
781   \begin{qstest}{\CheckAux}{\CheckAux}%
782     \def\GetFileContents##1{%
783       \pdfunescapehex{%
784         \pdffiledump length\pdffilesize{##1}{##1}%
785       }%
786     }%
787     \Expect*{\GetFileContents{\jobname.aux}}{%
788       *{\GetFileContents{\jobname-aux.aux}}%
789     }%
790 }
791 \g@addto@macro\@EndChecks{\@CheckAux}

```

```

792 \let\org@filec@ntents\filec@ntents
793 \def\filec@ntents{%
794   \catcode'\|=0 %
795   \org@filec@ntents
796 }
797 \begingroup
798   \endlinechar=-1 %
799   \catcode'\|=0 %
800   \@makeother\#%
801   \@makeother\|%
802   \gdef\AddList{%
803     \providecommand\AskInc@AddList[2]{%
804       \ifcase\ifx#1\@undefined
805         \else\ifx#1\relax
806         \else\ifx#1\@empty
807         \else1\fi\fi\fi0 %
808       \gdef#1{#2}%
809     \else
810       \g@addto@macro#1{,#2}%
811     \fi
812   }%
813 }%
814 \gdef\AddFile{%
815   \providecommand\AskInc@AddFile{%
816     \AskInc@AddList\AskInc@Files
817   }%
818 }%
819 \gdef\AddMatch{%
820   \providecommand\AskInc@AddMatch{%
821     \AskInc@AddList\AskInc@MatchFiles
822   }%
823 }%
824 \endgroup
825 \</test0>

```

3.4 First run tests

```

826 \<*test1>
827 \Stack{{} }
828 \CheckAnswer{*}
829 \begin{filecontents*}{\jobname.aux}
830 \relax
831 \end{filecontents*}
832 \begin{filecontents*}{\jobname-aux.aux}
833 \relax\space
834 \gdef\AskInc@Answer{*}
835 \gdef\AskInc@MatchMethod{}
836 \AddList
837 \AddFile
838 \AskInc@AddFile{askinclude-a}
839 \@input{askinclude-a.aux}
840 \AskInc@AddFile{askinclude-b}
841 \@input{askinclude-b.aux}
842 \AskInc@AddFile{askinclude-c}
843 \@input{askinclude-c.aux}
844 \@EndChecks
845 \end{filecontents*}
846 \begin{document}
847 \CheckInclude{a}{true}
848 \CheckInclude{b}{true}
849 \CheckInclude{c}{true}
850 \</test1>
851 \<*test2>

```

```

852 \Stack{*}
853 \CheckAnswer{*}
854 \begin{filecontents*}{\jobname.aux}
855 \relax
856 \end{filecontents*}
857 \begin{filecontents*}{\jobname-aux.aux}
858 \relax|space
859 \gdef\AskInc@Answer{*}
860 \gdef\AskInc@MatchMethod{}
861 |AddList
862 |AddFile
863 \AskInc@AddFile{askinclude-a}
864 \@input{askinclude-a.aux}
865 \AskInc@AddFile{askinclude-b}
866 \@input{askinclude-b.aux}
867 \AskInc@AddFile{askinclude-c}
868 \@input{askinclude-c.aux}
869 \@EndChecks
870 \end{filecontents*}
871 \begin{document}
872 \CheckInclude{a}{true}
873 \CheckInclude{b}{true}
874 \CheckInclude{c}{true}
875 </test2>

876 <*test3>
877 \Stack{-}%
878 \CheckAnswer{-}
879 \begin{filecontents*}{\jobname.aux}
880 \relax
881 \end{filecontents*}
882 \begin{filecontents*}{\jobname-aux.aux}
883 \relax|space
884 \gdef\AskInc@Answer{-}
885 \gdef\AskInc@MatchMethod{}
886 |AddList
887 |AddFile
888 \AskInc@AddFile{askinclude-a}
889 \@input{askinclude-a.aux}
890 \AskInc@AddFile{askinclude-b}
891 \@input{askinclude-b.aux}
892 \AskInc@AddFile{askinclude-c}
893 \@input{askinclude-c.aux}
894 \@EndChecks
895 \end{filecontents*}
896 \begin{document}
897 \CheckInclude{a}{false}
898 \CheckInclude{b}{false}
899 \CheckInclude{c}{false}
900 </test3>

901 <*test4>
902 \Stack{{askinclude-b}}
903 \CheckAnswer{askinclude-b}
904 \begin{filecontents*}{\jobname.aux}
905 \relax
906 \end{filecontents*}
907 \begin{filecontents*}{\jobname-aux.aux}
908 \relax|space
909 \gdef\AskInc@Answer{askinclude-b}
910 \gdef\AskInc@MatchMethod{}
911 |AddList
912 |AddFile
913 \AskInc@AddFile{askinclude-a}

```



```

914 \@input{askinclude-a.aux}
915 \AskInc@AddFile{askinclude-b}
916 \@input{askinclude-b.aux}
917 \AskInc@AddFile{askinclude-c}
918 \@input{askinclude-c.aux}
919 \@EndChecks
920 \end{filecontents*}
921 \begin{document}
922 \CheckInclude{a}{false}
923 \CheckInclude{b}{true}
924 \CheckInclude{c}{false}
925 </test4>

926 <*test5>
927 \Stack{{askinclude-a,askinclude-c}}
928 \CheckAnswer{askinclude-a,askinclude-c}
929 \begin{filecontents*}{\jobname.aux}
930 \relax
931 \end{filecontents*}
932 \begin{filecontents*}{\jobname-aux.aux}
933 \relax|space
934 \gdef\AskInc@Answer{askinclude-a,askinclude-c}
935 \gdef\AskInc@MatchMethod{}
936 |AddList
937 |AddFile
938 \AskInc@AddFile{askinclude-a}
939 \@input{askinclude-a.aux}
940 \AskInc@AddFile{askinclude-b}
941 \@input{askinclude-b.aux}
942 \AskInc@AddFile{askinclude-c}
943 \@input{askinclude-c.aux}
944 \@EndChecks
945 \end{filecontents*}
946 \begin{document}
947 \CheckInclude{a}{true}
948 \CheckInclude{b}{false}
949 \CheckInclude{c}{true}
950 </test5>

951 <*test6>
952 \Stack{?yny}
953 \CheckAnswer{askinclude-a,askinclude-c}
954 \begin{filecontents*}{\jobname.aux}
955 \relax
956 \end{filecontents*}
957 \begin{filecontents*}{\jobname-aux.aux}
958 \relax|space
959 \gdef\AskInc@Answer{}
960 \gdef\AskInc@MatchMethod{}
961 |AddList
962 |AddFile
963 \AskInc@AddFile{askinclude-a}
964 \gdef\AskInc@Answer{askinclude-a}
965 \@input{askinclude-a.aux}
966 \AskInc@AddFile{askinclude-b}
967 \@input{askinclude-b.aux}
968 \AskInc@AddFile{askinclude-c}
969 \gdef\AskInc@Answer{askinclude-a,askinclude-c}
970 \@input{askinclude-c.aux}
971 \@EndChecks
972 \end{filecontents*}
973 \begin{document}
974 \CheckInclude{a}{true}
975 \CheckInclude{b}{false}

```

```

976 \CheckInclude{c}{true}
977 </test6>
978 <*test7>
979 \Stack{?yA}
980 \CheckAnswer{askinclude-a,askinclude-b,askinclude-c}
981 \begin{filecontents*}{\jobname.aux}
982 \relax
983 \end{filecontents*}
984 \begin{filecontents*}{\jobname-aux.aux}
985 \relax\space
986 \gdef\AskInc@Answer{}
987 \gdef\AskInc@MatchMethod{}
988 |AddList
989 |AddFile
990 \AskInc@AddFile{askinclude-a}
991 \gdef\AskInc@Answer{askinclude-a}
992 \@input{askinclude-a.aux}
993 \AskInc@AddFile{askinclude-b}
994 \gdef\AskInc@Answer{askinclude-a,askinclude-b}
995 \@input{askinclude-b.aux}
996 \AskInc@AddFile{askinclude-c}
997 \gdef\AskInc@Answer{askinclude-a,askinclude-b,askinclude-c}
998 \@input{askinclude-c.aux}
999 \@EndChecks
1000 \end{filecontents*}
1001 \begin{document}
1002 \CheckInclude{a}{true}
1003 \CheckInclude{b}{true}
1004 \CheckInclude{c}{true}
1005 </test7>
1006 <*test8>
1007 \Stack{?yD}
1008 \CheckAnswer{askinclude-a,askinclude-b,askinclude-c}
1009 \begin{filecontents*}{\jobname.aux}
1010 \relax
1011 \end{filecontents*}
1012 \begin{filecontents*}{\jobname-aux.aux}
1013 \relax\space
1014 \gdef\AskInc@Answer{}
1015 \gdef\AskInc@MatchMethod{}
1016 |AddList
1017 |AddFile
1018 \AskInc@AddFile{askinclude-a}
1019 \gdef\AskInc@Answer{askinclude-a}
1020 \@input{askinclude-a.aux}
1021 \AskInc@AddFile{askinclude-b}
1022 \gdef\AskInc@Answer{askinclude-a,askinclude-b}
1023 \@input{askinclude-b.aux}
1024 \AskInc@AddFile{askinclude-c}
1025 \gdef\AskInc@Answer{askinclude-a,askinclude-b,askinclude-c}
1026 \@input{askinclude-c.aux}
1027 \@EndChecks
1028 \end{filecontents*}
1029 \begin{document}
1030 \CheckInclude{a}{true}
1031 \CheckInclude{b}{true}
1032 \CheckInclude{c}{true}
1033 </test8>
1034 <*test9>
1035 \Stack{?yN}
1036 \CheckAnswer{askinclude-a}
1037 \begin{filecontents*}{\jobname.aux}

```

```

1038 \relax
1039 \end{filecontents*}
1040 \begin{filecontents*}{\jobname-aux.aux}
1041 \relax\space
1042 \gdef\AskInc@Answer{}
1043 \gdef\AskInc@MatchMethod{}
1044 |AddList
1045 |AddFile
1046 \AskInc@AddFile{askinclude-a}
1047 \gdef\AskInc@Answer{askinclude-a}
1048 \@input{askinclude-a.aux}
1049 \AskInc@AddFile{askinclude-b}
1050 \@input{askinclude-b.aux}
1051 \AskInc@AddFile{askinclude-c}
1052 \@input{askinclude-c.aux}
1053 \@EndChecks
1054 \end{filecontents*}
1055 \begin{document}
1056 \CheckInclude{a}{true}
1057 \CheckInclude{b}{false}
1058 \CheckInclude{c}{false}
1059 </test9>

1060 <*test10>
1061 \Stack{?{}N}
1062 \CheckAnswer{askinclude-a}
1063 \begin{filecontents*}{\jobname.aux}
1064 \relax
1065 \end{filecontents*}
1066 \begin{filecontents*}{\jobname-aux.aux}
1067 \relax\space
1068 \gdef\AskInc@Answer{}
1069 \gdef\AskInc@MatchMethod{}
1070 |AddList
1071 |AddFile
1072 \AskInc@AddFile{askinclude-a}
1073 \gdef\AskInc@Answer{askinclude-a}
1074 \@input{askinclude-a.aux}
1075 \AskInc@AddFile{askinclude-b}
1076 \@input{askinclude-b.aux}
1077 \AskInc@AddFile{askinclude-c}
1078 \@input{askinclude-c.aux}
1079 \@EndChecks
1080 \end{filecontents*}
1081 \begin{document}
1082 \CheckInclude{a}{true}
1083 \CheckInclude{b}{false}
1084 \CheckInclude{c}{false}
1085 </test10>

1086 <*test11>
1087 \Stack{?xyxxN}
1088 \CheckAnswer{askinclude-a}
1089 \begin{filecontents*}{\jobname.aux}
1090 \relax
1091 \end{filecontents*}
1092 \begin{filecontents*}{\jobname-aux.aux}
1093 \relax\space
1094 \gdef\AskInc@Answer{}
1095 \gdef\AskInc@MatchMethod{}
1096 |AddList
1097 |AddFile
1098 \AskInc@AddFile{askinclude-a}
1099 \gdef\AskInc@Answer{askinclude-a}

```

```

1100 \@input{askinclude-a.aux}
1101 \AskInc@AddFile{askinclude-b}
1102 \@input{askinclude-b.aux}
1103 \AskInc@AddFile{askinclude-c}
1104 \@input{askinclude-c.aux}
1105 \@EndChecks
1106 \end{filecontents*}
1107 \begin{document}
1108 \CheckInclude{a}{true}
1109 \CheckInclude{b}{false}
1110 \CheckInclude{c}{false}
1111 \</test11>

```

3.5 Second run tests

```

1112 \<test12>
1113 \Stack{{}}
1114 \CheckAnswer{askinclude-b}
1115 \begin{filecontents*}{\jobname.aux}
1116 \gdef\AskInc@Answer{askinclude-b}
1117 \gdef\AskInc@MatchMethod{}
1118 |AddList
1119 |AddFile
1120 \AskInc@AddFile{askinclude-a}
1121 \AskInc@AddFile{askinclude-b}
1122 \AskInc@AddFile{askinclude-c}
1123 \end{filecontents*}
1124 \begin{filecontents*}{\jobname-aux.aux}
1125 \relax|space
1126 \gdef\AskInc@Answer{askinclude-b}
1127 \gdef\AskInc@MatchMethod{}
1128 |AddList
1129 |AddFile
1130 \AskInc@AddFile{askinclude-a}
1131 \@input{askinclude-a.aux}
1132 \AskInc@AddFile{askinclude-b}
1133 \@input{askinclude-b.aux}
1134 \AskInc@AddFile{askinclude-c}
1135 \@input{askinclude-c.aux}
1136 \@EndChecks
1137 \end{filecontents*}
1138 \begin{document}
1139 \CheckInclude{a}{false}
1140 \CheckInclude{b}{true}
1141 \CheckInclude{c}{false}
1142 \</test12>

1143 \<test13>
1144 \Stack{{}}
1145 \CheckAnswer{askinclude-b}
1146 \begin{filecontents*}{\jobname.aux}
1147 \gdef\AskInc@Answer{askinclude-b}
1148 \gdef\AskInc@MatchMethod{}
1149 |AddList
1150 |AddFile
1151 \AskInc@AddFile{askinclude-a}
1152 \AskInc@AddFile{askinclude-b}
1153 \end{filecontents*}
1154 \begin{filecontents*}{\jobname-aux.aux}
1155 \relax|space
1156 \gdef\AskInc@Answer{askinclude-b}
1157 \gdef\AskInc@MatchMethod{}
1158 |AddList
1159 |AddFile

```

```

1160 \AskInc@AddFile{askinclude-a}
1161 \@input{askinclude-a.aux}
1162 \AskInc@AddFile{askinclude-b}
1163 \@input{askinclude-b.aux}
1164 \AskInc@AddFile{askinclude-c}
1165 \@input{askinclude-c.aux}
1166 \@EndChecks
1167 \end{filecontents*}
1168 \begin{document}
1169 \CheckInclude{a}{false}
1170 \CheckInclude{b}{true}
1171 \CheckInclude{c}{false}
1172 </test13>

1173 <*test14>
1174 \Stack{{}}
1175 \CheckAnswer{*}
1176 \begin{filecontents*}{\jobname.aux}
1177 \gdef\AskInc@Answer{*}
1178 \gdef\AskInc@MatchMethod{}
1179 |AddList
1180 |AddFile
1181 \AskInc@AddFile{askinclude-a}
1182 \AskInc@AddFile{askinclude-b}
1183 \end{filecontents*}
1184 \begin{filecontents*}{\jobname-aux.aux}
1185 \relax|space
1186 \gdef\AskInc@Answer{*}
1187 \gdef\AskInc@MatchMethod{}
1188 |AddList
1189 |AddFile
1190 \AskInc@AddFile{askinclude-a}
1191 \@input{askinclude-a.aux}
1192 \AskInc@AddFile{askinclude-b}
1193 \@input{askinclude-b.aux}
1194 \AskInc@AddFile{askinclude-c}
1195 \@input{askinclude-c.aux}
1196 \@EndChecks
1197 \end{filecontents*}
1198 \begin{document}
1199 \CheckInclude{a}{true}
1200 \CheckInclude{b}{true}
1201 \CheckInclude{c}{true}
1202 </test14>

1203 <*test15>
1204 \Stack{?D}
1205 \CheckAnswer{askinclude-a,askinclude-c}
1206 \begin{filecontents*}{\jobname.aux}
1207 \gdef\AskInc@Answer{askinclude-a}
1208 \gdef\AskInc@MatchMethod{}
1209 |AddList
1210 |AddFile
1211 \AskInc@AddFile{askinclude-a}
1212 \AskInc@AddFile{askinclude-b}
1213 \end{filecontents*}
1214 \begin{filecontents*}{\jobname-aux.aux}
1215 \relax|space
1216 \gdef\AskInc@Answer{}
1217 \gdef\AskInc@MatchMethod{}
1218 |AddList
1219 |AddFile
1220 \AskInc@AddFile{askinclude-a}
1221 \gdef\AskInc@Answer{askinclude-a}

```

```

1222 \@input{askinclude-a.aux}
1223 \AskInc@AddFile{askinclude-b}
1224 \@input{askinclude-b.aux}
1225 \AskInc@AddFile{askinclude-c}
1226 \gdef\AskInc@Answer{askinclude-a,askinclude-c}
1227 \@input{askinclude-c.aux}
1228 \@EndChecks
1229 \end{filecontents*}
1230 \begin{document}
1231 \CheckInclude{a}{true}
1232 \CheckInclude{b}{false}
1233 \CheckInclude{c}{true}
1234 </test15>

1235 <*test16>
1236 \Stack{?A}
1237 \CheckAnswer{askinclude-a,askinclude-b,askinclude-c}
1238 \begin{filecontents*}{\jobname.aux}
1239 \gdef\AskInc@Answer{askinclude-a}
1240 \gdef\AskInc@MatchMethod{}
1241 |AddList
1242 |AddFile
1243 \AskInc@AddFile{askinclude-a}
1244 \AskInc@AddFile{askinclude-b}
1245 \end{filecontents*}
1246 \begin{filecontents*}{\jobname-aux.aux}
1247 \relax|space
1248 \gdef\AskInc@Answer{}
1249 \gdef\AskInc@MatchMethod{}
1250 |AddList
1251 |AddFile
1252 \AskInc@AddFile{askinclude-a}
1253 \gdef\AskInc@Answer{askinclude-a}
1254 \@input{askinclude-a.aux}
1255 \AskInc@AddFile{askinclude-b}
1256 \gdef\AskInc@Answer{askinclude-a,askinclude-b}
1257 \@input{askinclude-b.aux}
1258 \AskInc@AddFile{askinclude-c}
1259 \gdef\AskInc@Answer{askinclude-a,askinclude-b,askinclude-c}
1260 \@input{askinclude-c.aux}
1261 \@EndChecks
1262 \end{filecontents*}
1263 \begin{document}
1264 \CheckInclude{a}{true}
1265 \CheckInclude{b}{true}
1266 \CheckInclude{c}{true}
1267 </test16>

1268 <*test17>
1269 \Stack{?N}
1270 \CheckAnswer{}
1271 \begin{filecontents*}{\jobname.aux}
1272 \gdef\AskInc@Answer{askinclude-a}
1273 \gdef\AskInc@MatchMethod{}
1274 |AddList
1275 |AddFile
1276 \AskInc@AddFile{askinclude-a}
1277 \AskInc@AddFile{askinclude-b}
1278 \end{filecontents*}
1279 \begin{filecontents*}{\jobname-aux.aux}
1280 \relax|space
1281 \gdef\AskInc@Answer{}
1282 \gdef\AskInc@MatchMethod{}
1283 |AddList

```

```

1284 |AddFile
1285 \AskInc@AddFile{askinclude-a}
1286 \@input{askinclude-a.aux}
1287 \AskInc@AddFile{askinclude-b}
1288 \@input{askinclude-b.aux}
1289 \AskInc@AddFile{askinclude-c}
1290 \@input{askinclude-c.aux}
1291 \@EndChecks
1292 \end{filecontents*}
1293 \begin{document}
1294 \CheckInclude{a}{false}
1295 \CheckInclude{b}{false}
1296 \CheckInclude{c}{false}
1297 </test17>

1298 <*test18>
1299 \Stack{}
1300 \CheckAnswer{askinclude-a}
1301 \begin{filecontents*}{\jobname.aux}
1302 \gdef\AskInc@Answer{askinclude-a}
1303 \gdef\AskInc@MatchMethod{}
1304 |AddList
1305 |AddFile
1306 \AskInc@AddFile{askinclude-a}
1307 \AskInc@AddFile{askinclude-b}
1308 \end{filecontents*}
1309 \begin{filecontents*}{\jobname-aux.aux}
1310 \relax|space
1311 \gdef\AskInc@Answer{askinclude-a}
1312 \gdef\AskInc@MatchMethod{}
1313 |AddList
1314 |AddFile
1315 \AskInc@AddFile{askinclude-a}
1316 \@input{askinclude-a.aux}
1317 \AskInc@AddFile{askinclude-b}
1318 \@input{askinclude-b.aux}
1319 \AskInc@AddFile{askinclude-c}
1320 \@input{askinclude-c.aux}
1321 \@EndChecks
1322 \end{filecontents*}
1323 \begin{document}
1324 \CheckInclude{a}{false}
1325 \CheckInclude{b}{true}
1326 \CheckInclude{c}{false}
1327 </test18>

1328 <*test19>
1329 \Stack{?N}
1330 \CheckAnswer{}
1331 \begin{filecontents*}{\jobname.aux}
1332 \gdef\AskInc@Answer{askinclude-a}
1333 \gdef\AskInc@MatchMethod{}
1334 |AddList
1335 |AddFile
1336 \AskInc@AddFile{askinclude-a}
1337 \AskInc@AddFile{askinclude-b}
1338 \end{filecontents*}
1339 \begin{filecontents*}{\jobname-aux.aux}
1340 \relax|space
1341 \gdef\AskInc@Answer{}
1342 \gdef\AskInc@MatchMethod{}
1343 |AddList
1344 |AddFile
1345 \AskInc@AddFile{askinclude-a}

```

```

1346 \@input{askinclude-a.aux}
1347 \AskInc@AddFile{askinclude-b}
1348 \@input{askinclude-b.aux}
1349 \AskInc@AddFile{askinclude-c}
1350 \@input{askinclude-c.aux}
1351 \@EndChecks
1352 \end{filecontents*}
1353 \ifx\AtBeginDocumentMode Y%
1354   \AtBeginDocument{\includeonly{askinclude-b}}%
1355 \else
1356   \includeonly{askinclude-b}%
1357 \fi
1358 \begin{document}
1359 \CheckInclude{a}{false}
1360 \CheckInclude{b}{false}
1361 \CheckInclude{c}{false}
1362 \end{test19}

```

3.6 Compatibility for changing from old to new

```

1363 <*test20>
1364 \Stack{{}}
1365 \CheckAnswer{*}
1366 \begin{filecontents*}{\jobname.aux}
1367 \@includeonly{*}
1368 \end{filecontents*}
1369 \begin{filecontents*}{\jobname-aux.aux}
1370 \relax|space
1371 \gdef\AskInc@Answer{*}
1372 \gdef\AskInc@MatchMethod{}
1373 |AddList
1374 |AddFile
1375 \AskInc@AddFile{askinclude-a}
1376 \@input{askinclude-a.aux}
1377 \AskInc@AddFile{askinclude-b}
1378 \@input{askinclude-b.aux}
1379 \AskInc@AddFile{askinclude-c}
1380 \@input{askinclude-c.aux}
1381 \@EndChecks
1382 \end{filecontents*}
1383 \begin{document}
1384 \CheckInclude{a}{true}
1385 \CheckInclude{b}{true}
1386 \CheckInclude{c}{true}
1387 </test20>

1388 <*test21>
1389 \Stack{{}}
1390 \CheckAnswer{-}
1391 \begin{filecontents*}{\jobname.aux}
1392 \@includeonly{}
1393 \end{filecontents*}
1394 \begin{filecontents*}{\jobname-aux.aux}
1395 \relax|space
1396 \gdef\AskInc@Answer{-}
1397 \gdef\AskInc@MatchMethod{}
1398 |AddList
1399 |AddFile
1400 \AskInc@AddFile{askinclude-a}
1401 \@input{askinclude-a.aux}
1402 \AskInc@AddFile{askinclude-b}
1403 \@input{askinclude-b.aux}
1404 \AskInc@AddFile{askinclude-c}
1405 \@input{askinclude-c.aux}

```



```

1406 \@EndChecks
1407 \end{filecontents*}
1408 \begin{document}
1409 \CheckInclude{a}{false}
1410 \CheckInclude{b}{false}
1411 \CheckInclude{c}{false}
1412 </test21>

1413 <*test22>
1414 \Stack{{}}
1415 \CheckAnswer{askinclude-a,askinclude-c}
1416 \begin{filecontents*}{\jobname.aux}
1417 \@includeonly{askinclude-a,askinclude-c}
1418 \end{filecontents*}
1419 \begin{filecontents*}{\jobname-aux.aux}
1420 \relax|space
1421 \gdef\AskInc@Answer{askinclude-a,askinclude-c}
1422 \gdef\AskInc@MatchMethod{}
1423 |AddList
1424 |AddFile
1425 \AskInc@AddFile{askinclude-a}
1426 \@input{askinclude-a.aux}
1427 \AskInc@AddFile{askinclude-b}
1428 \@input{askinclude-b.aux}
1429 \AskInc@AddFile{askinclude-c}
1430 \@input{askinclude-c.aux}
1431 \@EndChecks
1432 \end{filecontents*}
1433 \begin{document}
1434 \CheckInclude{a}{true}
1435 \CheckInclude{b}{false}
1436 \CheckInclude{c}{true}
1437 </test22>

```

3.7 Regular expressions

```

1438 <*test23>
1439 \Stack{{\makematch}{*,!askinclude-b}}
1440 \CheckAnswer{*,!askinclude-b}
1441 \CheckMatchMethod{\makematch}
1442 \begin{filecontents*}{\jobname.aux}
1443 \relax
1444 \end{filecontents*}
1445 \begin{filecontents*}{\jobname-aux.aux}
1446 \relax|space
1447 \gdef\AskInc@Answer{*,!askinclude-b}
1448 \gdef\AskInc@MatchMethod{\makematch}
1449 |AddList
1450 |AddFile
1451 \AskInc@AddFile{askinclude-a}
1452 |AddMatch
1453 \AskInc@AddMatch{askinclude-a}
1454 \@input{askinclude-a.aux}
1455 \AskInc@AddFile{askinclude-b}
1456 \@input{askinclude-b.aux}
1457 \AskInc@AddFile{askinclude-c}
1458 \AskInc@AddMatch{askinclude-c}
1459 \@input{askinclude-c.aux}
1460 \@EndChecks
1461 \end{filecontents*}
1462 \begin{document}
1463 \CheckInclude{a}{true}
1464 \CheckInclude{b}{false}
1465 \CheckInclude{c}{true}

```

```

1466 </test23>
1467 <*test24>
1468 \Stack{\pdfmatch}{.,!askinclude-b}}
1469 \CheckAnswer{.,!askinclude-b}
1470 \CheckMatchMethod{\pdfmatch}
1471 \begin{filecontents*}{\jobname.aux}
1472 \relax
1473 \end{filecontents*}
1474 \begin{filecontents*}{\jobname-aux.aux}
1475 \relax\space
1476 \gdef\AskInc@Answer{.,!askinclude-b}
1477 \gdef\AskInc@MatchMethod{\pdfmatch}
1478 |AddList
1479 |AddFile
1480 \AskInc@AddFile{askinclude-a}
1481 |AddMatch
1482 \AskInc@AddMatch{askinclude-a}
1483 \@input{askinclude-a.aux}
1484 \AskInc@AddFile{askinclude-b}
1485 \@input{askinclude-b.aux}
1486 \AskInc@AddFile{askinclude-c}
1487 \AskInc@AddMatch{askinclude-c}
1488 \@input{askinclude-c.aux}
1489 \@@EndChecks
1490 \end{filecontents*}
1491 \begin{document}
1492 \CheckInclude{a}{true}
1493 \CheckInclude{b}{false}
1494 \CheckInclude{c}{true}
1495 </test24>

```

3.8 Common end

```

1496 <*test0>
1497 \end{document}
1498 </test0>

```

4 Installation

4.1 Download

Package. This package is available on CTAN¹:

CTAN:macros/latex/contrib/oberdiek/askinclude.dtx The source file.

CTAN:macros/latex/contrib/oberdiek/askinclude.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:macros/latex/contrib/oberdiek/oberdiek-tds.zip

TDS refers to the standard “A Directory Structure for T_EX 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
```

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

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-`TEX`:

```
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 `TEX` distribution (`teTEX`, `mikTEX`, ...) relies on file name databases, you must refresh these. For example, `teTEX` 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 L^AT_EX. The .dtx chooses its action depending on the format:

plain-T_EX: Run docstrip and extract the files.

L^AT_EX: Generate the documentation.

If you insist on using L^AT_EX for docstrip (really, docstrip does not need L^AT_EX), 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 pdfL^AT_EX:

```
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
- [3] David Kastrup: *makematch.sty: Pattern matching with wildcards*; 2007/02/23;
[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](#).

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`).

[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`.
- 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.

8 Index

Numbers written in *italic* refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in roman refer to the code lines where the entry is used.

Symbols		<code>\-</code>	332
<code>\!</code>	335	<code>\.</code>	203, 346
<code>\#</code>	677, 800	<code>\:</code>	345
<code>\\$</code>	200, 337	<code>\?</code>	197, 333
<code>\%</code>	680	<code>\@</code>	673
<code>\(</code>	193, 338	<code>\@CheckAnswer</code>	762, 764
<code>\)</code>	194, 339	<code>\@CheckAux</code>	780, 791
<code>*</code>	195, 331	<code>\@CheckMatchMethod</code>	772, 774
<code>\+</code>	196, 340	<code>\@EndChecks</code>	752,
<code>\,</code>	334		756, 762, 772, 791, 844, 869,

894, 919, 944, 971, 999, 1027, 1053, 1079, 1105, 1136, 1166, 1196, 1228, 1261, 1291, 1321, 1351, 1381, 1406, 1431, 1460, 1489	\AskInc@CheckMatch 453, 455
\@Stack 709, 717, 720, 722, 730, 732, 737	\AskInc@QuestionHeader 559, 573, 574
\@StackEmpty 709, 729, 730, 737	\AskInc@AddFile 72, 79, 85, 815, 838, 840, 842, 863, 865, 867, 888, 890, 892, 913, 915, 917, 938, 940, 942, 963, 966, 968, 990, 993, 996, 1018, 1021, 1024, 1046, 1049, 1051, 1072, 1075, 1077, 1098, 1101, 1103, 1120, 1121, 1122, 1130, 1132, 1134, 1151, 1152, 1160, 1162, 1164, 1181, 1182, 1190, 1192, 1194, 1211, 1212, 1220, 1223, 1225, 1243, 1244, 1252, 1255, 1258, 1276, 1277, 1285, 1287, 1289, 1306, 1307, 1315, 1317, 1319, 1336, 1337, 1345, 1347, 1349, 1375, 1377, 1379, 1400, 1402, 1404, 1425, 1427, 1429, 1451, 1455, 1457, 1480, 1484, 1486
\@car 717	\AskInc@AddList 48, 73, 503, 803, 816, 821
\@cdr 722	\AskInc@AddMatch 502, 509, 515, 820, 1453, 1458, 1482, 1487
\@empty 53, 115, 118, 131, 175, 206, 218, 227, 231, 242, 243, 253, 269, 285, 297, 354, 367, 370, 381, 382, 388, 391, 395, 415, 453, 480, 488, 527, 537, 562, 582, 806	\AskInc@Answer . 103, 114, 115, 116, 212, 214, 216, 247, 251, 258, 292, 371, 381, 386, 397, 410, 520, 562, 563, 565, 568, 766, 834, 859, 884, 909, 934, 959, 964, 969, 986, 991, 994, 997, 1014, 1019, 1022, 1025, 1042, 1047, 1068, 1073, 1094, 1099, 1116, 1126, 1147, 1156, 1177, 1186, 1207, 1216, 1221, 1226, 1239, 1248, 1253, 1256, 1259, 1272, 1281, 1302, 1311, 1332, 1341, 1371, 1396, 1421, 1447, 1476
\@firstofone 34, 436, 447	\AskInc@AskFile 91, 95, 383
\@firstoftwo 254, 538	\AskInc@AskQuestion 560, 571
\@for 246, 258, 259, 542, 543, 552	\AskInc@AskResult 561, 580, 582, 583, 586, 589, 592, 594, 597, 599, 602, 604, 608
\@gobble 80, 91, 92, 436, 449, 510	\AskInc@AtBeginDocument 33, 38
\@ifundefined 101, 106, 129, 211, 217, 221, 230, 523, 526	\AskInc@AtEnd 8, 9, 138, 621
\@includeonly 113, 1367, 1392, 1417	\AskInc@CharAll 185, 592
\@input 839, 841, 843, 864, 866, 868, 889, 891, 893, 914, 916, 918, 939, 941, 943, 965, 967, 970, 992, 995, 998, 1020, 1023, 1026, 1048, 1050, 1052, 1074, 1076, 1078, 1100, 1102, 1104, 1131, 1133, 1135, 1161, 1163, 1165, 1191, 1193, 1195, 1222, 1224, 1227, 1254, 1257, 1260, 1286, 1288, 1290, 1316, 1318, 1320, 1346, 1348, 1350, 1376, 1378, 1380, 1401, 1403, 1405, 1426, 1428, 1430, 1454, 1456, 1459, 1483, 1485, 1488	\AskInc@CharAsterisk 178, 212, 247, 387, 524, 531
\@mainaux 40, 756	\AskInc@CharDefault 187, 602
\@makeoother 331, 332, 333, 334, 335, 336, 337, 338, 339, 340, 341, 342, 343, 344, 345, 346, 800, 801	\AskInc@CharExclam 182, 457
\@ne 658, 665	\AskInc@CharHash 45, 49, 51, 53, 60, 62
\@nil 453, 455, 717, 722	\AskInc@CharMinus 179, 250, 251, 390, 535
\@partlist 415, 416, 418, 568	\AskInc@CharNo 184, 534, 550, 554, 589, 599
\@secondoftwo 256, 540	\AskInc@CharNone 186, 597
\@spaces 267, 292, 312, 318, 320, 321, 322, 348	\AskInc@CharPlus 180, 248, 262, 268
\@undefined 49, 427, 438, 497, 804	\AskInc@CharQuestion 181, 380
\[. 191, 341, 678	\AskInc@CharYes 183, 532, 546, 551, 561, 586, 594
\. 190, 343, 674, 801	\AskInc@CheckMatch 445, 452
\{ 201, 675	\AskInc@Default 532, 534, 546, 550, 551, 554, 577, 583, 604
\} 202, 676	
\] 192, 342, 679	
\^ 199, 336	
\ 198, 344, 794, 799	
_ 681	
A	
\advance 658, 665	
\answer 347, 350, 572, 575, 580, 581	
\AnswerText 761, 767	
\AskInc@Answer 350, 353, 356, 359, 370, 371, 380, 386, 387, 390	

[illegible]

<code>\CompatibilityMode</code>	642	<code>\ifAskInc@PdfMatchAvailable</code>	
<code>\count@</code>	651, 655, 657, 658, 661, 663, 664, 665	<code>\ifcase</code>	153, 162, 225, 303, 360
<code>\csname</code>	3, 155, 748	<code>\ifnum</code>	657, 664
D		<code>\ifx</code>	49, 51, 53, 115, 131, 155, 224, 243, 247, 251, 253, 261, 268, 269, 285, 312, 353, 356, 359, 370, 380, 387, 390, 395, 415, 422, 457, 486, 488, 531, 535, 537, 545, 550, 553, 561, 562, 582, 586, 589, 592, 597, 602, 642, 690, 709, 737, 748, 804, 805, 806, 1353
<code>\DeclareOption</code>	125, 126, 127, 159, 163, 167, 174	<code>\immediate</code>	40, 756
<code>\do</code>	246, 260, 544, 552	<code>\include</code>	90, 93, 245, 745
<code>\documentclass</code>	645	<code>\includeonly</code>	122, 142, 143, 636, 1354, 1356
<code>\documentstyle</code>	643	<code>\IncludeTests</code>	704
E		J	
<code>\end</code> 712, 742, 750, 768, 778, 789, 831, 845, 856, 870, 881, 895, 906, 920, 931, 945, 956, 972, 983, 1000, 1011, 1028, 1039, 1054, 1065, 1080, 1091, 1106, 1123, 1137, 1153, 1167, 1183, 1197, 1213, 1229, 1245, 1262, 1278, 1292, 1308, 1322, 1338, 1352, 1368, 1382, 1393, 1407, 1418, 1432, 1444, 1461, 1473, 1490, 1497	<code>\jobname</code>	787, 788, 829, 832, 854, 857, 879, 882, 904, 907, 929, 932, 954, 957, 981, 984, 1009, 1012, 1037, 1040, 1063, 1066, 1089, 1092, 1115, 1124, 1146, 1154, 1176, 1184, 1206, 1214, 1238, 1246, 1271, 1279, 1301, 1309, 1331, 1339, 1366, 1369, 1391, 1394, 1416, 1419, 1442, 1445, 1471, 1474	
<code>\endcsname</code>	3, 155, 748	L	
<code>\endinput</code>	139	<code>\LogTests</code>	705
<code>\endlinechar</code>	798	<code>\loop</code>	652, 662
<code>\EndTest</code>	668, 701	M	
<code>\escapechar</code>	190	<code>\makeatletter</code>	649, 706
<code>\Expect</code> ...	711, 736, 747, 766, 776, 787	<code>\makeatother</code>	685
F		<code>\MakeMatcher</code>	433
<code>\filea</code>	625	<code>\MatchMethodText</code>	771, 777
<code>\fileb</code>	629	<code>\MessageBreak</code> .	122, 146, 169, 364, 608
<code>\filec</code>	633	N	
<code>\filec@ntents</code>	792, 793	<code>\NeedsTeXFormat</code>	2
G		<code>\newcommand</code> .	89, 141, 731, 744, 760, 770
<code>\g@addto@macro</code>	62, 272, 418, 491, 565, 762, 772, 791, 810	<code>\newif</code>	153, 441
<code>\gdef</code>	60, 103, 108, 114, 116, 134, 416, 520, 563, 593, 598, 603, 761, 771, 808, 834, 835, 859, 860, 884, 885, 909, 910, 934, 935, 959, 960, 964, 969, 986, 987, 991, 994, 997, 1014, 1015, 1019, 1022, 1025, 1042, 1043, 1047, 1068, 1069, 1073, 1094, 1095, 1099, 1116, 1117, 1126, 1127, 1147, 1148, 1156, 1157, 1177, 1178, 1186, 1187, 1207, 1208, 1216, 1217, 1221, 1226, 1239, 1240, 1248, 1249, 1253, 1256, 1259, 1272, 1273, 1281, 1282, 1302, 1303, 1311, 1312, 1332, 1333, 1341, 1342, 1371, 1372, 1396, 1397, 1421, 1422, 1447, 1448, 1476, 1477	O	
<code>\GetFileContents</code>	782, 787, 788	<code>\org@filec@ntents</code>	792, 795
I		<code>\org@typein</code>	707, 713
<code>\if@filesw</code>	39, 755	P	
<code>\if@partsw</code>	120	<code>\PackageWarning</code>	144, 363
<code>\ifAskInc@Found</code>	441, 446	<code>\PackageWarningNoLine</code>	121, 168, 475, 607
		<code>\pdffiledump</code>	784
		<code>\pdffilesize</code>	784
		<code>\pdfmatch</code> .	170, 305, 365, 458, 465, 476
		<code>\pdfunescapehex</code>	783
		<code>\ProcessOptions</code>	128, 177
		<code>\providecommand</code>	36, 48, 72, 88, 113, 502, 803, 815, 820
		<code>\ProvidesPackage</code>	31

R		\the 4, 10, 655
\RangeCatcodeInvalid		\TMP@EnsureCode 7,
. 660, 669, 670, 671, 672		14, 15, 16, 17, 18, 19, 20, 21,
\renewcommand 93, 143		22, 23, 24, 25, 26, 27, 28, 29, 30
\repeat 659, 666		\typein 347, 575, 707, 708
\RequirePackage 409, 430		\typeout 235, 236, 237, 238,
\RestoreCatcodes . . 650, 653, 654, 684		239, 245, 267, 279, 283, 292,
		293, 296, 308, 311, 317, 318,
		320, 321, 322, 349, 579, 725, 726
S		
\space 122,		
145, 206, 298, 305, 312, 318, 348		
\Stack 731, 827, 852,		U
877, 902, 927, 952, 979, 1007,		\unexpanded 716, 721, 766, 767, 776, 777
1035, 1061, 1087, 1113, 1144,		\usepackage 646, 647, 692, 696
1174, 1204, 1236, 1269, 1299,		
1329, 1364, 1389, 1414, 1439, 1468		W
		\write 40, 756
T		
\Test 668, 687		Z
		\zap@space 205