

The `asymptote` package

John Bowman, Tom Prince, and Will Robertson

2010/09/30 v1.22

Abstract

This package provides integration of inline and external Asymptote graphics within a \LaTeX document.

Contents

1 Introduction

This is the documentation for the \LaTeX package `asymptote` which accompanies the Asymptote drawing package. For further details on Asymptote, please see its documentation in `asymptote.pdf`.

2 User syntax

2.1 Package loading and options

The package may take two options at load time: `inline` or `attach`. These options can also be set at any time with the `\asysetup{<options>}` command, or specified individually in the optional argument to each `asy` environment or `asyinclude` command.

The `inline` option uses Asymptote's 'inline' mode whereby included graphics have their labels typeset in the environment of the document they are contained within. Otherwise the Asymptote graphics are self-contained and their formatting is independent of the document.

The `attach` option allows generated graphics to be embedded within the PDF using the `attachfile2` package; please load that package separately if you wish to use it. The `attach` option takes precedence over the `inline` option.

This package produces quite a number of output files, which by default are created in the same directory as the \LaTeX document that is being compiled. To keep things more tidy, you can specify an output directory for these files by defining the `\asydir` command. For example, if you wish to store the figure files in the subdirectory `asytmp/`, then you would write `\renewcommand\asydir{asytmp}`.

Alternatively (and tentatively), you may write `dir=asytmp` in either the `asy` environment options or the options to `\asysetup`.

2.2 Commands for inserting Asymptote graphics

The main environment defined by the package is the `asy` environment, in which verbatim Asymptote code is placed that will be compiled for generating a graphic in the document. For example,

```
\begin{figure}
\begin{asy}[ <options> ]
<ASYMPTOTE CODE>
\end{asy}
\caption{...}\label{...}
```

If you have Asymptote code in a separate file, you can include it with the `\asyinclude[<options>]{<filename>}` command.

For Asymptote code that should be included in *every* graphic, define it using the `asydef` environment.

2.3 Graphics options

Both the `asy` environment and the `\asyinclude` command take optional parameters for controlling aspects of the graphics creation. In addition to locally setting `inline` and `attach`, the following options may also be used:

height Height of the figure

width Width of the figure

viewportheight Viewport height for 3D figures

viewportwidth Viewport width for 3D figures

These may also be set globally using the `\asysetup` command.

3 Processing the document

After running `LATEX` on the document, it is necessary to process the Asymptote graphics so they can be included in the next compilation. The simplest procedure is a recipe such as

```
pdflatex mydoc
asy mydoc-*.asy
pdflatex mydoc
```

This technique will recompile each graphic every time, however. To only recompile graphics that have changed, use the `latexmk` tool. Asymptote is distributed with a `latexmkrc` configuration file; place this file in a place where `latexmk` will find it and your document may be compiled, including the `asy` compilations, with `latexmk mydoc` or `latexmk --pdf mydoc`.

4 Implementation

```
1 \def\Asymptote{\tt Asymptote}
2 \InputIfFileExists{\jobname.pre}{-}{-}
```

4.1 Allocations

Allocations

```
3 \newbox\ASYbox
4 \newcounter{asy}
5 \newwrite\AsyStream
6 \newwrite\AsyPreStream
7 \newif\ifASYinline
8 \newif\ifASYattach
```

4.2 Packages

```
9 \RequirePackage{keyval}
10 \RequirePackage{ifthen}
11 \RequirePackage{color,graphicx}
```

Emulating packages We cannot assume that Asymptote users have recent T_EX distributions. (E.g., Fedora until recently still shipped t_EX.) So load ifpdf and ifxetex if they exist; otherwise, emulate them.

In due course, delete this code and just load the packages.

```
12 \IfFileExists{ifpdf.sty}{
13   \RequirePackage{ifpdf}
14 }{
15   \expandafter\newif\csname ifpdf\endcsname
16   \ifx\pdfoutput\@undefined\else
17     \ifcase\pdfoutput\else
18       \pdftrue
19     \fi
20   \fi
21 }

22 \IfFileExists{ifxetex.sty}{
23   \RequirePackage{ifxetex}
24 }{
25   \expandafter\newif\csname ifxetex\endcsname
26   \ifx\XeTeXversion\@undefined\else
27     \xetextrue
28   \fi
29 }
```

\CatchFileDef Used for \asyinclude.

```
30 \IfFileExists{catchfile.sty}{
31   \RequirePackage{catchfile}
32 }
```

```

33 \newcommand\CatchFileDef[3]{%
34   \begingroup
35   \everyeof{%
36     \ENDCATCHFILEMARKER
37     \noexpand
38   }%
39   \long\def\@tempa##1\ENDCATCHFILEMARKER{%
40     \endgroup
41     \def#1{##1}%
42   }%
43   #3%
44   \expandafter\@tempa\@input #2\relax
45 }
46 }

```

Ensuring attachfile2 is loaded if [attach] is requested

```

47 \newif\if@asy@attachfile@loaded
48 \AtBeginDocument{%
49   \@ifpackageloaded{attachfile2}{\@asy@attachfile@loadedtrue}{}%
50   \let\asy@check@attachfile\asy@check@attachfile@loaded
51 }
52 \newcommand\asy@check@attachfile@loaded{%
53   \if@asy@attachfile@loaded\else
54     \PackageError{asymptote}{You must load the attachfile2 package}{^^J%
55       You have requested the [attach] option for some or all of your^^J%
56       Asymptote graphics, which requires the attachfile2 package.^^J%
57       Please load it in the document preamble.^^J%
58     }%
59   \fi
60 }
61 \newcommand\asy@check@attachfile{%
62   \AtBeginDocument{\asy@check@attachfile@loaded}%
63   \let\asy@check@attachfile\@empty
64 }

```

Macros

```

65 \def\csarg#1#2{\expandafter#1\csname#2\endcsname}

```

4.3 Package options

```

66 \DeclareOption{inline}{%
67   \ASYinlinetrue
68 }
69 \DeclareOption{attach}{%
70   \asy@check@attachfile
71   \ASYattachtrue
72 }

```

```

73 \ProcessOptions*
74 \def\asydir{}
75 \def\ASYprefix{}

```

4.4 Testing for PDF output

Note this is not quite the same as `\ifpdf`, since we still want PDF output when using XeTeX.

```

76 \newif\ifASYPDF
77 \ifxetex
78   \ASYPDFtrue
79 \else
80   \ifpdf
81     \ASYPDFtrue
82   \fi
83 \fi
84 \ifASYPDF
85   \def\AsyExtension{pdf}
86 \else
87   \def\AsyExtension{eps}
88 \fi

```

4.5 Bug squashing

```

89 \def\unquoteJobname#1"#2"#3\relax{%
90   \def\rawJobname{#1}%
91   \ifx\rawJobname\empty
92     \def\rawJobname{#2}%
93   \fi
94 }
95 \expandafter\unquoteJobname\jobname""\relax

```

Work around jobname bug in MiKTeX 2.5 and 2.6: Turn stars in file names (resulting from spaces, etc.) into minus signs

```

96 \def\fixstar#1*#2\relax{%
97   \def\argtwo{#2}%
98   \ifx\argtwo\empty
99     \gdef\Jobname{#1}%
100   \else
101     \fixstar#1-#2\relax
102   \fi
103 }
104 \expandafter\fixstar\rawJobname*\relax

```

Work around bug in dvips.def: allow spaces in file names.

```

105 \def\Ginclude@eps#1{%
106   \message{<#1>}%
107   \bgroup
108   \def\@tempa{!}%
109   \dimen@\Gin@req@width
110   \dimen@ii.1bp\relax

```

```

111 \divide\dimen@\dimen@ii
112 \@tempdima\Gin@req@height
113 \divide\@tempdima\dimen@ii
114 \special{PSfile=#1\space
115 llx=\Gin@llx\space
116 lly=\Gin@lly\space
117 urx=\Gin@urx\space
118 ury=\Gin@ury\space
119 \ifx\Gin@scalex\@tempa\else rwi=\number\dimen@\space\fi
120 \ifx\Gin@scaley\@tempa\else rhi=\number\@tempdima\space\fi
121 \ifGin@clip clip\fi}%
122 \egroup
123 }

```

4.6 Input/Output

```

124 \immediate\openout\AsyPreStream=\jobname.pre\relax
125 \AtEndDocument{\immediate\closeout\AsyPreStream}

126 \def\WriteAsyLine#1{%
127 \immediate\write\AsyStream{\detokenize{#1}}%
128 }

129 \def\globalASYdefs{}
130 \def\WriteGlobalAsyLine#1{%
131 \expandafter\g@addto@macro
132 \expandafter\globalASYdefs
133 \expandafter{\detokenize{#1^^J}}%
134 }

```

4.7 Commands for verbatim processing environments

```

135 \def\ProcessAsymptote#1{%
136 \begingroup
137 \def\CurrentAsymptote{#1}%
138 \let\do\@makeother \dospecials
139 \@makeother\^^L and whatever other special cases
140 \endlinechar'\^^M \catcode'\^^M=12 \xAsymptote
141 }

```

Need lots of comment chars here because *line end* is no longer a space character.

```

142 \begingroup
143 \catcode'\^^M=12 \endlinechar=-1\relax%
144 \gdef\xAsymptote{%
145 \expandafter\ProcessAsymptoteLine%
146 }
147 \gdef\ProcessAsymptoteLine#1^^M{%
148 \def\@tempa{#1}%
149 {%
150 \escapechar=-1\relax%
151 \xdef\@tempb{\string\end\string\{\CurrentAsymptote\string\}}%
152 }%
153 \ifx\@tempa\@tempb%

```

```

154     \edef\next{\endgroup\noexpand\end{\CurrentAsymptote}}%
155     \else%
156         \ThisAsymptote{#1}%
157         \let\next\ProcessAsymptoteLine%
158     \fi%
159     \next%
160 }
161 \endgroup

```

4.8 User interface

```

162 \newcommand\asy[1][]{%
163     \stepcounter{asy}%
164     \setkeys{ASYkeys}{#1}%

```

Disable the "inline" option if "attach" is enabled:

```

165     \ifASYattach
166         \ASYinlinefalse
167     \fi
168     \ifx\asydir\empty\else
169         \def\ASYprefix{\asydir/}%
170     \fi
171     \immediate\write\AsyPreStream{%
172         \noexpand\InputIfFileExists{%
173             \ASYprefix\noexpand\jobname-\the\c@asy.pre}{\}{\}%
174     }
175     \asy@write@graphic@header
176     \let\ThisAsymptote\WriteAsyLine
177     \ProcessAsymptote{asy}%
178 }

179 \def\endasy{%
180     \asy@finalise@stream
181     \asy@input@graphic
182 }

183 \def\asy@write@graphic@header{%
184     \immediate\openout\AsyStream=\ASYprefix\jobname-\the\c@asy.asy\relax
185     \gdef\AsyFile{\ASYprefix\Jobname-\the\c@asy}%
186     \immediate\write\AsyStream{%
187         if(!settings.multipleView) settings.batchView=false;^^J%
188         \ifxetex
189             settings.tex="xelatex";^^J%
190         \else\ifASYPDF
191             settings.tex="pdflatex";^^J%
192         \fi\fi
193         \ifASYinline
194             settings.inlinetex=true;^^J%
195             deletetopreamble();^^J%
196         \fi
197         defaultfilename="\Jobname-\the\c@asy";^^J%
198         if(settings.render < 0) settings.render=4;^^J%

```

```

199     settings.outformat="";^^J%
200     \ifASYattach
201         settings.inlineimage=false;^^J%
202         settings.embed=false;^^J%
203         settings.toolbar=true;^^J%
204     \else
205         settings.inlineimage=true;^^J%
206         settings.embed=true;^^J%
207         settings.toolbar=false;^^J%
208         viewportmargin=(2,2);^^J%
209     \fi
210     \globalASYdefs
211 }%
212 }

```

```

213 \def\asy@finalise@stream{%

```

Setting `size()`. Only inserted if one of the dimensions is set explicitly (i.e., if both height and width are not empty).

```

214     \ifx\ASYwidth\@empty
215         \ifx\ASYheight\@empty
216             % write nothing!
217         \else
218             \immediate\write\AsyStream{size(0,\ASYheight);}%
219         \fi
220     \else
221         \ifx\ASYheight\@empty
222             \immediate\write\AsyStream{size(\ASYwidth,0);}%
223         \else
224             \immediate\write\AsyStream{size(\ASYwidth,\ASYheight);}%
225         \fi
226     \fi

```

Setting `viewportsize()`. Same logic as for `size()`. Note that `\ASYviewportwidth` varies according to `\ASYattach`, by default, so it needs to be fully expanded before checking if it equals zero.

```

227     \edef\ASYviewportwidth{\ASYviewportwidth}% locally "freeze" viewportwidth
228     \ifx\ASYviewportwidth\@empty
229         \ifx\ASYviewportheight\@empty
230             % write nothing!
231         \else
232             \immediate\write\AsyStream{viewportsize=(0,\ASYviewportheight);}%
233         \fi
234     \else
235         \ifx\ASYviewportheight\@empty
236             \immediate\write\AsyStream{viewportsize=(\ASYviewportwidth,0);}%
237         \else
238             \immediate\write\AsyStream{%
239                 viewportsize=(\ASYviewportwidth,\ASYviewportheight);}%
240         \fi
241     \fi

```



```

242 \immediate\closeout\AsyStream
243 }
244 \def\asy@input@graphic{%
245   \ifASYinline
246     \IfFileExists{"\AsyFile.tex"}{%
247       \catcode'::=12\relax
248       \@input"\AsyFile.tex"\relax
249     }{%
250       \PackageWarning{asymptote}{file '\AsyFile.tex' not found}%
251     }%
252   \else
253     \IfFileExists{"\AsyFile.\AsyExtension"}{%
254       \ifASYattach
255         \ifASYPDF
256           \IfFileExists{"\AsyFile+0.pdf"}{%
257             \setbox\ASYbox=\hbox{\includegraphics[hiresbb]{"\AsyFile+0".pdf}}%
258           }{%
259             \setbox\ASYbox=\hbox{\includegraphics[hiresbb]{"\AsyFile".pdf}}%
260           }%
261         \else
262           \setbox\ASYbox=\hbox{\includegraphics[hiresbb]{"\AsyFile.eps"}}%
263         \fi
264         \textattachfile{\AsyFile.\AsyExtension}{\phantom{\copy\ASYbox}}%
265         \vskip-\ht\ASYbox
266         \indent
267         \box\ASYbox
268       \else
269         \ifASYPDF
270           \includegraphics[hiresbb]{"\AsyFile".pdf}%
271         \else
272           \includegraphics[hiresbb]{"\AsyFile.eps"}%
273         \fi
274       \fi
275     }{%
3D PRC figures require inline mode.
276     \IfFileExists{"\AsyFile.tex"}{%
277       \catcode'::=12
278       \@input"\AsyFile.tex"\relax
279     }{%
280       \PackageWarning{asymptote}{%
281         file '\AsyFile.\AsyExtension' not found%
282       }%
283     }%
284   }%
285 \fi
286 }
287 \def\asydef{%
288   \let\ThisAsymptote\WriteGlobalAsyLine
289   \ProcessAsymptote{asydef}%

```

```

290 }

291 \newcommand\asyinclude[2][]{\%
292   \begingroup
293   \stepcounter{asy}%
294   \setkeys{ASYkeys}{#1}%
295   \ifASYattach
296     \ASYinlinefalse
297   \fi
298   \ifx\asydir\empty\else
299     \def\ASYprefix{\asydir/}%
300   \fi
301   \immediate\write\AsyPreStream{%
302     \noexpand\inputIfFileExists{%
303       \ASYprefix\noexpand\jobname-\the\c@asy.pre}{-}{-}%
304   }
305   \asy@write@graphic@header
306   \IfFileExists{#2.asy}{\%
307     \CatchFileDef\@tempa{#2.asy}{\%
308       \let\do\@makeother
309       \dospecials
310       \endlinechar=10\relax
311     }
312   }{
313     \IfFileExists{#2}{\%
314       \CatchFileDef\@tempa{#2}{\%
315         \let\do\@makeother
316         \dospecials
317         \endlinechar=10\relax
318       }
319     }{
320       \PackageWarning{asymptote}{file #2 not found}%
321     }
322   }
323   \immediate\write\AsyStream{\unexpanded\expandafter{\@tempa}}%
324   \asy@finalise@stream
325   \asy@input@graphic
326   \endgroup
327 }

328 \newcommand{\ASYanimategraphics}[5][]{\%
329   \IfFileExists{_#3.pdf}{\%
330     \animategraphics[#1][#2]{_#3}{#4}{#5}%
331   }{}
332 }

```

4.9 Keys for graphics processing

```

333 \newcommand\asysetup[1]{\setkeys{ASYkeys}{#1}}
334 \define@key{ASYkeys}{dir}{\%
335   \def\asydir{#1}%

```

```

336 }
337 \def\ASYwidth{}
338 \define@key{ASYkeys}{width}{%
339   \edef\ASYwidth{\the\dimexpr#1\relax}%
340 }
341 \def\ASYheight{}
342 \define@key{ASYkeys}{height}{%
343   \edef\ASYheight{\the\dimexpr#1\relax}%
344 }
345 \def\ASYviewportwidth{\ASYdefaultviewportwidth}
346 \def\ASYdefaultviewportwidth{%
347   \ifASYattach \@empty \else \the\linewidth \fi
348 }
349 \define@key{ASYkeys}{viewportwidth}{%
350   \edef\ASYviewportwidth{\the\dimexpr#1\relax}%
351 }
352 \def\ASYviewportheight{}
353 \define@key{ASYkeys}{viewportheight}{%
354   \edef\ASYviewportheight{\the\dimexpr#1\relax}%
355 }

356 \define@key{ASYkeys}{attach}[true]{%
357   \ifthenelse{\equal{#1}{true}}
358     {\ASYattachtrue}
359     {\ASYattachfalse}%
360 }
361 \define@key{ASYkeys}{inline}[true]{%
362   \ifthenelse{\equal{#1}{true}}
363     {\ASYinlinetrue}
364     {\ASYinlinefalse}%
365 }

```