The dottex package*

Lars Kotthoff
metalhead@metalhead.ws

April 2, 2006

1 Introduction

This package allows you to include dot and neato graphs in your LaTeX documents.

The dot/neato code is extracted from the document and written to `.dot`/`.neato` files. Then, if shell escape is used, the graph files are automatically processed and converted to PostScript files, which will then be included. If shell escape isn't used, the user will have to manually convert the files.

Shell escape is available in the web2c TeX compiler, it allows the execution of shell code during the compilation of a TeX document. It's disabled by default, you'll have to edit your configuration files or give the `-shell-escape` option to `latex`. A check is performed whether shell escape really works, so you might get warnings that the `.dot`/`.neato` files need to be converted manually although you enabled shell escape.

2 Requirements

To use dottex, you'll need the `graphicx`, `keyval` and `moreverb` packages, ps2epsi and epstopdf if you want to use the pdf option, and of course, dot / neato.

3 Usage

To load the package, simply `\usepackage{dottex}` in your document preamble. Options that can be passed to the package are

`[`⟨***shell***⟩`]` Use shell escape to automatically generate the graphs from the dot source files. This is the default.

`[`⟨***noshell***⟩`]` Don't use shell escape, graphs must be generated manually.

`[`⟨***miktex***⟩`]` We're using mikTeX.

`[`⟨***pdf***⟩`]` PDF output, generate .pdf files of the graphs out of .ps.

The following environments can be used to include graphs:

`dotpic` Within this environment, you can specify arbitrary dot code, for example

*This document corresponds to dottex v0.5, dated 2006/04/02.

```
a -> b;
b -> a;.
```

The `digraph` preamble and the surrounding braces are inserted automatically.

`neatopic` Within this environment, you can specify arbitrary neato code, for example

```
a -- b;
b -- a;.
```

The `graph` preamble and the surrounding braces are inserted automatically.

For both environments, you may specify width or height of the picture using [width=⟨*width*⟩] or [height=⟨*height*⟩] as in the `includegraphics` command.

4 Implementation

4.1 Initialization

```
\newif\ifShellEscape
\newif\ifmiktex \miktexfalse
\newif\ifpdf \pdffalse

\DeclareOption{shell}{\ShellEscapetrue}
\DeclareOption{noshell}{\ShellEscapefalse}
\DeclareOption{miktex}{\global\miktextrue}
\DeclareOption{pdf}{\pdftrue}

\ExecuteOptions{shell}
\ProcessOptions\relax
%% test if shell escape really works
\ifShellEscape
    \def\tmpfile{/tmp/w18-test-\the\year\the\month\the\day\the\time}
    \ifmiktex
        \immediate\write18{rem >"\tmpfile"}
    \else
        \immediate\write18{touch \tmpfile}
    \fi
    \ifmiktex
        \IfFileExists{\tmpfile.}{\ShellEscapetrue}{\ShellEscapefalse}
    \else
        \IfFileExists{\tmpfile}{\ShellEscapetrue}{\ShellEscapefalse}
    \fi
\fi

\ifShellEscape
    \PackageInfo{dottex}
    {Automatically converting dot/neato files}
\else
    \PackageWarningNoLine{dottex}
        {Shell escape not enabled.\MessageBreak
        You'll need to convert the graphs yourself.}
\fi
\newcounter{fignum}
```

4.2 `.dot`/`.neato` write out

```
\def\figname{\jobname-dottex-fig\thefignum}
```

```
37
38 \def\dotverbatimwrite#1{%
39     \def\BeforeStream
40     {\message{Opening Dot stream=\figname.dot}%
41         \immediate\write\verbatim@out{\string digraph\space G\space {/*}*/}
42     }
43     \@bsphack
44     \immediate\openout \verbatim@out #1
45     \BeforeStream%
46     \let\do\@makeother\dospecials
47     \catcode`\^^M\active
48     \def\verbatim@processline{%
49         \immediate\write\verbatim@out
50         {\the\verbatim@line}}%
51     \verbatim@start}
52 \def\enddotverbatimwrite{%
53     \immediate\write\verbatim@out{/*{*/}}
54     \immediate\closeout\verbatim@out
55     \@esphack}
56
57 \def\neatoverbatimwrite#1{%
58     \def\BeforeStream
59     {\message{Opening Neato stream=\figname.neato}%
60         \immediate\write\verbatim@out{\string graph\space G\space {/*}*/}
61     }
62     \@bsphack
63     \immediate\openout \verbatim@out #1
64     \BeforeStream%
65     \let\do\@makeother\dospecials
66     \catcode`\^^M\active
67     \def\verbatim@processline{%
68         \immediate\write\verbatim@out
69         {\the\verbatim@line}}%
70     \verbatim@start}
71 \def\endneatoverbatimwrite{%
72     \immediate\write\verbatim@out{/*{*/}}
73     \immediate\closeout\verbatim@out
74     \@esphack}
```

The spurious braces (commented out in the `.dot`/`.neato` file) are necessary because LaTeX gets confused with only one brace.

4.3 Environment definition

```
75 \define@key{pic}{width}{\def\dotwidth{#1}}
76 \define@key{pic}{height}{\def\dotheight{#1}}
77 \newenvironment{dotpic}[1][]{\stepcounter{fignum}%
78 \let\dotwidth\undefined
79 \let\dotheight\undefined
80 \setkeys{pic}{#1}
81     \xdef\dotCutFile{\figname.dot}
82     \dotverbatimwrite{\dotCutFile}}
83     {\enddotverbatimwrite%
84     \dotgraphicsinclude}
85
86 \newenvironment{neatopic}[1][]{\stepcounter{fignum}%
```

```
\let\dotwidth\undefined
\let\dotheight\undefined
\setkeys{pic}{#1}
    \xdef\neatoCutFile{\figname.neato}
    \neatoverbatimwrite{\neatoCutFile}}
    {\endneatoverbatimwrite%
    \neatographicsinclude}
```

4.4 .dot/.neato file processing

```
\long\gdef\dotgraphicsprocess{%
    \ifShellEscape
\IfFileExists{\figname.dot}{%
\immediate\write18{dot -Tps -o \figname.ps \figname.dot}
\IfFileExists{\figname.ps}{%
\ifpdf
\immediate\write18{ps2epsi \figname.ps \figname.eps}
\immediate\write18{epstopdf \figname.eps}
\IfFileExists{\figname.pdf}{%
\PackageInfo{dottex}
{\figname.dot converted}}
{\PackageWarningNoLine{dottex}
{Conversion of \figname.dot failed.}}
\else
\PackageInfo{dottex}
{\figname.dot converted}
\fi}
{\PackageWarningNoLine{dottex}
{Conversion of \figname.dot failed.}}}{}
\fi}

\long\gdef\neatographicsprocess{%
    \ifShellEscape
        \IfFileExists{\figname.neato}{%
            \immediate\write18{neato -Tps -o \figname.ps \figname.neato}
            \IfFileExists{\figname.ps}{%
\ifpdf
\immediate\write18{ps2epsi \figname.ps \figname.eps}
\immediate\write18{epstopdf \figname.eps}
\IfFileExists{\figname.pdf}{%
\PackageInfo{dottex}
{\figname.dot converted}}
{\PackageWarningNoLine{dottex}
{Conversion of \figname.dot failed.}}
\else
\PackageInfo{dottex}
{\figname.neato converted}
\fi}
                {\PackageWarningNoLine{dottex}
                    {Conversion of \figname.neato failed.}}}{}
\fi}
```

4.5 Graph inclusion

```
\long\gdef\dotgraphicsinclude{\dotgraphicsprocess%
\ifpdf
```

```
\IfFileExists{\figname.pdf}{%
\ifx\dotwidth\undefined
\ifx\dotheight\undefined
\includegraphics{\figname}
\else
\includegraphics[height=\dotheight]{\figname}
\fi
\else
\ifx\dotheight\undefined
\includegraphics[width=\dotwidth]{\figname}
\else
\includegraphics[width=\dotwidth,height=\dotheight]{\figname}
\fi
\fi
}
{\PackageWarningNoLine{dottex}
{Please convert \figname.dot manually}}
\else
\IfFileExists{\figname.ps}{%
\ifx\dotwidth\undefined
\ifx\dotheight\undefined
\includegraphics{\figname}
\else
\includegraphics[height=\dotheight]{\figname}
\fi
\else
\ifx\dotheight\undefined
\includegraphics[width=\dotwidth]{\figname}
\else
\includegraphics[width=\dotwidth,height=\dotheight]{\figname}
\fi
\fi
}
{\PackageWarningNoLine{dottex}
{Please convert \figname.dot manually}}
\fi
}

\long\gdef\neatographicsinclude{\neatographicsprocess%
\ifpdf
\IfFileExists{\figname.pdf}{%
\ifx\dotwidth\undefined
\ifx\dotheight\undefined
\includegraphics{\figname}
\else
\includegraphics[height=\dotheight]{\figname}
\fi
\else
\ifx\dotheight\undefined
\includegraphics[width=\dotwidth]{\figname}
\else
\includegraphics[width=\dotwidth,height=\dotheight]{\figname}
\fi
\fi
```

```
}
{\PackageWarningNoLine{dottex}
{Please convert \figname.neato manually}}
\else
\IfFileExists{\figname.ps}{%
\ifx\dotwidth\undefined
\ifx\dotheight\undefined
\includegraphics{\figname}
\else
\includegraphics[height=\dotheight]{\figname}
\fi
\else
\ifx\dotheight\undefined
\includegraphics[width=\dotwidth]{\figname}
\else
\includegraphics[width=\dotwidth,height=\dotheight]{\figname}
\fi
\fi
}
{\PackageWarningNoLine{dottex}
{Please convert \figname.neato manually}}
\fi
}
```