

## 0 graphviz

graphviz is a freely available package for doing automated graph layout from AT&T Research. ('Free' here means economically free and also under some variety of open source licence, that is different from the GPL and the BSD licences.) The graphviz package includes the dot program, which reads a textual description of a graph and produces a graphical rendering of it. Many different graphics formats, include postscript, are supported.

There are two main web pages for the graphviz project:

- <http://www.graphviz.org>
- <http://www.research.att.com/sw/tools/graphviz/>

## 1 graphviz.tex

graphviz.tex contains some simple macros that allow one to write graphs in the dot/graphviz language inside a L<sup>A</sup>T<sub>E</sub>X document. It's easy. Here's how:

```
\usepackage[dvips]{graphicx}
\input{graphviz}
\digraph[scale=0.5]{MyGraph}{rankdir=LR; a->b; b->c}
```

### 1.1 the arguments to \digraph

The \digraph command takes three arguments:

1. parameters to the \includegraphics command that will include the postscript file of the graph [this is optional]: eg, 'scale=0.5'
2. the name of the graph; a file name.dot is created, and a file name.ps is expected to be produced from dot: eg, 'MyGraph'
3. the graph, specified in the dot/graphviz language:  
eg, 'rankdir=LR; a->b; b->c'

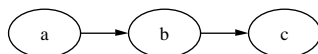
### 1.2 partial result

When you run L<sup>A</sup>T<sub>E</sub>X on a document with a \digraph command in it, it produces a dot file. You need to run dot to turn this into a postscript file. If you haven't done this yet, you'll get the following message in your document instead of your graph:

```
The file MyGraph.ps hasn't been created from MyGraph.dot yet.
Run 'dot -Tps -o MyGraph.ps MyGraph.dot' to create it.
Here is a bash loop to process all dot files in the current directory:
  for f in *.dot do ; dot -Tps -o ${f%.dot}.ps $f ; done
```

### 1.3 final result

After you run dot to create the postscript file, the next time you run L<sup>A</sup>T<sub>E</sub>X your graph will be included in your document. Here's our example:



## A Source listing of graphviz.tex

```
% graphviz.tex
% originally written by Derek Rayside, November 2003
% following an idea that Daniel Jackson implemented in his Tagger program
%
% parameters to \digraph:
% 1 - parameters for \includegraphics (optional; default value is "scale=1")
% 2 - name of the digraph
% 3 - body of the digraph

\newcommand{\digraph}[3][scale=1]{
  \newwrite\dotfile
  \immediate\openout\dotfile=#2.dot
  \immediate\write\dotfile{digraph #2 {\string#3}}
  \immediate\closeout\dotfile
  \IfFileExists{#2.ps}
  { % the postscript exists: include it
    { \includegraphics[#1]{#2} }
  } % the postscript doesn't exist: tell the user how to create it
  { \fbox{ \begin{tabular}{l}
    The file \texttt{#2.ps} hasn't been created from
    \texttt{#2.dot} yet. \\
    Run '\texttt{dot -Tps -o #2.ps #2.dot}' to create it. \\
    Here is a \textsf{bash} loop to process all \textsf{dot} files
    in the current directory: \\
    \texttt{
      for f in *.dot do ;
      dot -Tps -o \${f%.dot}.ps \$f ;
      done
    }
    \end{tabular}}
  }
}
```