

Dynamic Diagrams in Stata

E. F. Haghish

25 Jun 2016

Syntax

```
diagram [DOT | using filename] [, replace export(filename) magnify(real)
phantomjs(str) engine(name) ]
```

options	Description
replace	replace the exported diagram
engine(name)	specifies the graphViz engine for rendering the diagram which can be dot, osage, circo, neato, twopi and fdps. The default engine is dot
export(filename)	export the diagram. The file extension specifies the format and it can be .pdf, .png, .jpeg, .gif, or .bmp
magnify(real)	increases the resolution of the exported image by multiplying its resolution to the specified number. The value of the real number should be above 0 and by default is 1.0
phantomjs(str)	specifies the path to executable [phantomjs software](http://www.phantomjs.org/download.html) on the machine

Description

diagram renders [graphViz](#) graphs within Stata and exports them to several graphical formats including **pdf**, **png**, **jpeg**, **gif**, and **bmp**. This package is independent of the software and does not require installing graphViz. The **diagram** command can render a graph using *DOT* markup or by using file that includes the markup. For large graphs, it is advised to create a file and then render the graph.

[graphViz](#) is an open source graph visualization software which can be used to represent structural information such as diagrams of algorithms, groups, abstract graphs, and networks. The software has had notable applications in a variety of fields such as network visualization, bioinformatics, machine learning. The software renders graphics using a markup language which is highly customizable and can be altered with precision. Yet, it can be written in a very simple and basic way to make it human-readable. For more information regarding the software visit [graphViz homepage](#).

This package can have plenty of applications for Stata users. For example, it can be used to develop analysis diagrams, visualize information/algorithms, create diagrams for education purpose as well as write Stata programs that generate dynamic diagrams based on the results of data analysis.

Engines

[graphViz](#) has several engines which are **dot**, **neato**, **fdp**, **twopi**, **circo**, and **osage**. These engines render the diagrams differently but their markup is not identical. All of these engines are supported in this package but the user should read the engines carefully. The most popular engines are **dot** and **neato**. A brief description of the engines is presented below :

[dot](#) - "directed graphs" which is the default engine for rendering graphs where edges have directionality e.g. A -> B.

[neato](#) is recommended for undirected diagrams, especially when the size of the diagram is about 100 nodes or less.

[fdp](#) draws undirected graphs similar to **neato**, but applies different layouts.

[twopi](#) applies radial layouts.

[circo](#) applies circular layouts.

[osage](#) applies clustered layouts.

Third-party software

For exporting graphical files, the package requires [phantomJS](#), which is an open-source freeware available for Windows, Mac, and Linux. The path to the executable *phantomjs* file is required in order to export the graphical files. However, if the executable file is installed in the default local directory (e.g. /usr/local/bin/ in Mac), the *phantomjs(str)* can be ignored.

Example(s)

```
rendering DOT markup
  . graphviz digraph G {a -> b;}, magnify(2.5) export(..diagram.png)    ///
  phantomjs("/usr/local/bin/phantomjs")

rendering a graphviz file
  . graphviz using myfile.dot, magnify(2.5) export(..diagram.png)      ///
  phantomjs("/usr/local/bin/phantomjs")
```

Acknowledgements

The JavaScript engine of the program was developed by [Michael Daines](#).

Author

E. F. Haghish

Center for Medical Biometry and Medical Informatics

University of Freiburg, Germany

and

Department of Mathematics and Computer Science

University of Southern Denmark

haghish@imbi.uni-freiburg.de

<http://www.haghish.com/markdoc>

Package Updates on [Twitter](#)