

graph export — Export current graph[Description
Reference](#)[Quick start
Also see](#)[Syntax](#)[Options](#)[Remarks and examples](#)

Description

`graph export` exports to a file the graph displayed in a Graph window.

Quick start

Save the graph displayed in the Graph window as `mygraph.pdf`
`graph export mygraph.pdf`

Same as above
`graph export mygraph, as(pdf)`

Save the graph named “MyGraph” in memory to disk as an EPS file
`graph export mygraph.eps, name(MyGraph)`

As above, and replace `mygraph.eps` if it exists
`graph export mygraph.eps, name(MyGraph) replace`

Save the graph displayed in the Graph window as `mygraph.png` specifying the graph width and height in pixels
`graph export mygraph.png, width(600) height(450)`

Save the current graph as `mygraph.ps` setting the top and left margins to half an inch
`graph export mygraph.ps, tmargin(.5) lmargin(.5)`

As above, but suppress the Stata logo
`graph export mygraph.ps, tmargin(.5) lmargin(.5) logo(off)`

Specify that graph be printed on a page in landscape format
`graph export mygraph.ps, orientation(landscape)`

Convert the graph saved to disk as `mygraph.gph` to a PDF
`graph use mygraph`
`graph export mygraph.pdf`

Syntax

```
graph export newfilename.suffix [ , options ]
```

<i>options</i>	Description
<code>name(<i>windowname</i>)</code>	name of Graph window to export
<code>as(<i>fileformat</i>)</code>	desired format of output
<code>replace</code>	<i>newfilename</i> may already exist
<code>override_options</code>	override defaults in conversion

If `as()` is not specified, the output format is determined by the suffix of *newfilename.suffix*:

<i>suffix</i>	Implied option	Output format
<code>.ps</code>	<code>as(ps)</code>	PostScript
<code>.eps</code>	<code>as(eps)</code>	EPS (Encapsulated PostScript)
<code>.svg</code>	<code>as(svg)</code>	SVG (Scalable Vector Graphics)
<code>.wmf</code>	<code>as(wmf)</code>	Windows Metafile
<code>.emf</code>	<code>as(emf)</code>	Windows Enhanced Metafile
<code>.pdf</code>	<code>as(pdf)</code>	PDF (Portable Document Format)
<code>.png</code>	<code>as(png)</code>	PNG (Portable Network Graphics)
<code>.tif</code>	<code>as(tif)</code>	TIFF (Tagged Image File Format)
<i>other</i>		must specify <code>as()</code>

`ps`, `eps`, and `svg` are available for all versions of Stata; `png` and `tif` are available for all versions of Stata except Stata(console); and `wmf` and `emf` are available only for Stata for Windows.

<i>override_options</i>	Description
<code>ps_options</code>	when exporting to <code>ps</code>
<code>eps_options</code>	when exporting to <code>eps</code>
<code>svg_options</code>	when exporting to <code>svg</code>
<code>tif_options</code>	when exporting to <code>tif</code>
<code>png_options</code>	when exporting to <code>png</code>

There are no *override_options* for the `pdf` format.

Options

`name(windowname)` specifies which window to export from when exporting a graph. Omitting the `name()` option exports the topmost graph (Stata for Unix(GUI) users: see [Technical note for Stata for Unix\(GUI\) users](#)). The name for a window is displayed inside parentheses in the window title. For example, if the title for a Graph window is **Graph (MyGraph)**, the name for the window is **MyGraph**. If a graph is an **asis** or **graph7** graph where there is no name in the window title, specify "" for *windowname*.

`as(fileformat)` specifies the file format to which the graph is to be exported. This option is rarely specified because, by default, `graph export` determines the format from the suffix of the file being created.

`replace` specifies that it is okay to replace *filename.suffix* if it already exists.

override_options modify how the graph is converted. See [G-3] *ps_options*, [G-3] *eps_options*, [G-3] *svg_options*, [G-3] *tif_options*, and [G-3] *png_options*. See also [G-2] **graph set** for permanently setting default values for the *override_options*.

Remarks and examples

[stata.com](http://www.stata.com)

Graphs are exported by displaying them on the screen and then typing

```
. graph export filename.suffix
```

Remarks are presented under the following headings:

Exporting the graph displayed in a Graph window

Exporting a graph stored on disk

Exporting a graph stored in memory

If your interest is simply in printing a graph, see [G-2] **graph print**.

Exporting the graph displayed in a Graph window

There are three ways to export the graph displayed in a Graph window:

1. Right-click on the Graph window, select **Save Graph...**, and choose the appropriate **Save as type**.
2. Select **File > Save Graph...**, and choose the appropriate **Save as type**.
3. Type “`graph export filename.suffix`” in the Command window. Stata for Unix(GUI) users should use the `name()` option if there is more than one graph displayed to ensure that the correct graph is exported (see *Technical note for Stata for Unix(GUI) users*).

All three are equivalent. The advantage of `graph export` is that you can include it in do-files:

```
. graph ...                               (draw a graph)
. graph export filename.suffix (and export it)
```

By default, `graph export` determines the output type by the *suffix*. If we wanted to create an Encapsulated PostScript file, we might type

```
. graph export figure57.eps
```

Exporting a graph stored on disk

To export a graph stored on disk, type

```
. graph use gph_filename
. graph export output_filename.suffix
```

Do not specify `graph use`'s `nodraw` option; see [G-2] **graph use**.

Stata for Unix(console) users: follow the instructions just given, even though you have no Graph window and cannot see what has just been “displayed”. Use the graph, and then export it.

Exporting a graph stored in memory

To export a graph stored in memory but not currently displayed, type

```
. graph display name
. graph export filename.suffix
```

Do not specify `graph display`'s `nodraw` option; see [G-2] **graph display**.

Stata for Unix(console) users: follow the instructions just given, even though you have no Graph window and cannot see what has just been “displayed”. Display the graph, and then export it.

□ Technical note

Stata for Unix(GUI) users should note that X-Windows does not have a concept of a window z-order, which prevents Stata from determining which window is the topmost window. Instead, Stata determines which window is topmost based on which window has the focus. However, some window managers will set the focus to a window without bringing the window to the top. What Stata considers the topmost window may not appear topmost visually. For this reason, you should always use the `name()` option to ensure that the correct Graph window is exported.



Reference

Huber, C. 2014. How to create animated graphics using Stata. *The Stata Blog: Not Elsewhere Classified*. <http://blog.stata.com/2014/03/24/how-to-create-animated-graphics-using-stata/>.

Also see

[G-3] ***eps_options*** — Options for exporting to Encapsulated PostScript

[G-3] ***png_options*** — Options for exporting to portable network graphics (PNG) format

[G-3] ***ps_options*** — Options for exporting or printing to PostScript

[G-3] ***svg_options*** — Options for exporting to Scalable Vector Graphics

[G-3] ***tif_options*** — Options for exporting to tagged image file format (TIFF)

[G-2] **graph set** — Set graphics options

[G-2] **graph display** — Display graph stored in memory

[G-2] **graph print** — Print a graph

[G-2] **graph use** — Display graph stored on disk