

**graph describe** — Describe contents of graph in memory or on disk

[Description](#)      [Quick start](#)      [Menu](#)      [Syntax](#)  
[Remarks and examples](#)      [Stored results](#)      [Also see](#)

## Description

`graph describe` describes the contents of a graph in memory or a graph stored on disk.

## Quick start

Describe graph `mygraph` stored in memory

```
graph describe mygraph
```

Describe graph saved to disk as `mygraph.gph`

```
graph describe mygraph.gph
```

Same as above

```
graph describe "mygraph"
```

## Menu

Graphics > Manage graphs > Describe graph

## Syntax

```
graph describe [name]
```

<i>name</i>	Description
<i>simplename</i>	name of graph in memory
<i>filename.gph</i>	name of graph on disk
" <i>filename</i> "	name of graph on disk

If *name* is not specified, the graph currently displayed in the Graph window is described.

## Remarks and examples

See [\[G-2\] graph manipulation](#) for an introduction to the graph manipulation commands.

`graph describe` describes the contents of a graph, which may be stored in memory or on disk.

Without arguments, the graph stored in memory named `Graph` is described:

```
. use http://www.stata-press.com/data/r14/auto
(1978 Automobile Data)
. scatter mpg weight
(graph omitted)
```

```
. graph describe
Graph stored in memory
  name: Graph
  format: live
  created: 9 May 2014 14:26:12
  scheme: default
  size: 4 x 5.5
  dta file: auto.dta dated 13 Apr 2014 17:45
  command: twoway scatter mpg weight
```

In the above, the size is reported as  $y\text{size} \times x\text{size}$ , not the other way around.

When you type a name ending in `.gph`, the disk file is described:

```
. graph save myfile
. graph describe myfile.gph
myfile.gph stored on disk
  name: myfile.gph
  format: live
  created: 9 May 2014 14:26:12
  scheme: default
  size: 4 x 5.5
  dta file: auto.dta dated 13 Apr 2014 17:45
  command: twoway scatter mpg weight
```

If the file is saved in `asis` format—see [G-4] **concept: gph files**—only the name and format are listed:

```
. graph save picture, asis
. graph describe picture.gph
picture.gph stored on disk
  name: picture.gph
  format: asis
```

## Stored results

`graph describe` stores the following in `r()`:

Macros

<code>r(fn)</code>	<i>filename</i> or <i>filename.gph</i>
<code>r(ft)</code>	“old”, “asis”, or “live”

and, if `r(ft) == “live”`,

Macros

<code>r(command)</code>	command
<code>r(family)</code>	subcommand; <code>twoway</code> , <code>matrix</code> , <code>bar</code> , <code>dot</code> , <code>box</code> , or <code>pie</code>
<code>r(command_date)</code>	date on which command was run
<code>r(command_time)</code>	time at which command was run
<code>r(scheme)</code>	scheme name
<code>r(yssize)</code>	<code>ysize()</code> value
<code>r(xssize)</code>	<code>xsize()</code> value
<code>r(dtfile)</code>	<code>.dta</code> file in memory at <code>command_time</code>
<code>r(dtfile_date)</code>	<code>.dta</code> file date

Any of `r(command)`, `...`, `r(dtfile_date)` may be undefined, so refer to contents by using macro quoting.

## Also see

[G-2] [graph manipulation](#) — Graph manipulation commands

[G-2] [graph dir](#) — List names of graphs in memory and on disk