

## NAME

`dot` – preprocessor for drawing directed graphs

## SYNOPSIS

`dot` [`-Gname=value`] [`-Nname=value`] [`-Ename=value`] [`-Tlang`] [`-l libfile`] [`-o outfile`] [files]

## DESCRIPTION

`dot` draws directed graphs. It works well on DAGs and other graphs that can be drawn as hierarchies. It reads attributed graph files and writes drawings. By default, the output format `dot` is the input file with layout coordinates appended. To generate PostScript, use the `-Tps` option. Other choices are `-Tsvg` `-Tsvgz` (Structured Vector Graphics), `-Tfig` (XFIG graphics), `-Tmif` (FrameMaker graphics), `-Thppl` (HP pen plotters), and `-Tpcl` (Laserjet printers), `-Tpng` `-Tjpg` `-Tgif` (pixel graphics), `-Timap` (imagemap files for httpd servers that mark out rectangles for each node that has a non-null "href" attribute.), `-Tcmapx` (client-side imagemap for use in html and xhtml). (There are also two deprecated imagemap formats: `-Timap` and `-Tcmap`.)

Here is a synopsis of the graph file language.

**digraph** *name* { *statement-list* } is the top level graph. Statements may be:

*name*=*val*;

**node** [*name*=*val*];

**edge** [*name*=*val*]; Set default graph, node, or edge attribute *name* to *val*. Any subgraph, node, or edge appearing after this inherits the new default attributes.

**n0** [*name0*=*val0*,*name1*=*val1*,...]; Creates node **n0** (if it does not already exist) and sets its attributes according to the optional list.

**n0** -> **n1** -> ... -> **nn** [*name0*=*val0*,*name1*=*val1*,...]; Creates edges between nodes **n0**, **n1**, ..., **nn** and sets their attributes according to the optional list. Creates nodes as necessary.

**subgraph** *name* { *statement-list* } Creates a subgraph. Subgraphs may be used in place of **n0**, ..., **nn** in the above statements to create edges. **subgraph** *name* is optional; if missing, the subgraph is assigned an internal name.

Comments may be */\*C-like\*/* or *//C++-like*.

Attribute names and values are ordinary (C-style) strings. The following sections describe attributes that control graph layout.

## GRAPH ATTRIBUTES

**size**=*"x,y"* sets bounding box of drawing in inches.

**page**=*"x,y"* sets the PostScript pagination unit.

**ratio**=*f* sets the aspect ratio to *f* which may be a floating point number, or one of the keywords **fill**, **compress**, or **auto**.

**margin**=*f* sets the page margin (included in the page size).

**nodesep**=*f* sets the minimum separation between nodes.

**ranksep**=*f* sets the minimum separation between ranks.

**ordering**=**out** constrains order of out-edges in a subgraph according to their file sequence.

**rankdir**=**LR**|**RL**|**BT** requests a left-to-right, right-to-left, or bottom-to-top, drawing.

**pagedir**=[**TBLR**][**TBLR**] sets the major and minor order of pagination.

**rank**=**same** (or **min** or **max**) in a subgraph constrains the rank assignment of its nodes. If a subgraph's name has the prefix **cluster**, its nodes are drawn in a distinct rectangle of the layout. Clusters may be nested.

**rotate**=**90** sets landscape mode. (**orientation**=**land** is backward compatible but obsolete.)

**center**=*n* a non-zero value centers the drawing on the page.

**nslimit**=*f* or **mclimit**=*f* adjusts the bound on the number of network simplex or mincross iterations by the given ratio. For example, **mclimit**=**2.0** runs twice as long.

**layers**="id:id:id:id" is a sequence of layer identifiers for overlay diagrams. The PostScript array variable *layercolorseq* sets the assignment of colors to layers. The least index is 1 and each element must be a 3-element array to be interpreted as a color coordinate.

**color**=*colorvalue* sets foreground color (**bgcolor** for background).

**href**="url" the default url for image map files; in PostScript files, the base URL for all relative URLs, as recognized by Acrobat Distiller 3.0 and up.

**URL**="url" ("URL" is a synonym for "href".)

**stylesheet**="file.css" includes a reference to a stylesheet in -Tsvg and -Tsvgz outputs. Ignored by other formats.

## NODE ATTRIBUTES

**height**=*d* or **width**=*d* sets minimum height or width. Adding **fixedsize**=**true** forces these to be the actual size (text labels are ignored).

**shape**=**record polygon epsf** *builtin\_polygon*

*builtin\_polygon* is one of: **plaintext ellipse circle egg triangle box diamond trapezium parallelogram house hexagon octagon**. (Polygons are defined or modified by the following node attributes: **regular**, **peripheries**, **sides**, **orientation**, **distortion** and **skew**.) **epsf** uses the node's **shapefile** attribute as the path name of an external EPSF file to be automatically loaded for the node shape.

**label**=*text* where *text* may include escaped newlines \n, \l, or \r for center, left, and right justified lines. The string '\N' value will be replaced by the node name. Record labels may contain recursive box lists delimited by { | }. Port identifiers in labels are set off by angle brackets < >. In the graph file, use colon (such as, **node0:port28**).

**fontsize**=*n* sets the label type size to *n* points.

**fontname**=*name* sets the label font family name.

**color**=*colorvalue* sets the outline color, and the default fill color if **style**=filled and **fillcolor** is not specified.

**fillcolor**=*colorvalue* sets the fill color when **style**=filled. If not specified, the fillcolor when **style**=filled defaults to be the same as the outline color.

**fontcolor**=*colorvalue* sets the label text color.

A *colorvalue* may be "*h,s,v*" (hue, saturation, brightness) floating point numbers between 0 and 1, or an X11 color name such as **white black red green blue yellow magenta cyan** or **burlywood**, or a "#rrggb" (*red, green, blue, 2 hex characters each*) value.

**style**=**filled solid dashed dotted bold invis** or any Postscript code.

**layer**=*id* or *id:id* or "all" sets the node's active layers. The empty string means no layers (invisible).

The following attributes apply only to polygon shape nodes:

**regular**=*n* if *n* is non-zero then the polygon is made regular, i.e. symmetric about the x and y axis, otherwise the polygon takes on the aspect ratio of the label. *builtin\_polygons* that are not already regular are made regular by this attribute. *builtin\_polygons* that are already regular are not affected (i.e. they cannot be made asymmetric).

**peripheries**=*n* sets the number of periphery lines drawn around the polygon. This value supercedes the number of periphery lines of *builtin\_polygons*.

**sides**=*n* sets the number of sides to the polygon. *n*<3 results in an ellipse. This attribute is ignored by *builtin\_polygons*.

**orientation**=*f* sets the orientation of the first apex of the polygon counterclockwise from the vertical, in degrees. *f* may be a floating point number. The orientation of labels is not affected by this attribute. This attribute is added to the initial orientation of *builtin\_polygons*.

**distortion**=*f* sets the amount of broadening of the top and narrowing of the bottom of the polygon (relative to its orientation). Floating point values between -1 and +1 are suggested. This attribute is ignored by *builtin\_polygons*.

**skew**=*f* sets the amount of right-displacement of the top and left-displacement of the bottom of the polygon (relative to its orientation). Floating point values between -1 and +1 are suggested. This attribute is ignored by *builtin\_polygons*.

**href**="url" sets the url for the node in imagemap, PostScript and SVG files. The substring '\N' is substituted in the same manner as for the node label attribute.

**URL**="url" ("URL" is a synonym for "href".)

**target**="target" is a target string for client-side imagemaps and SVG, effective when nodes have a URL. The target string is used to determine which window of the browser is used for the URL. Setting it to "\_graphviz" will open a new window if it doesn't already exist, or reuse it if it does. If the target string is empty, the default, then no target attribute is included in the output. The substring '\N' is substituted in the same manner as for the node label attribute.

**tooltip**="tooltip" is a tooltip string for client-side imagemaps and SVG, effective when nodes have a URL. The tooltip string defaults to be the same as the label string, but this attribute permits nodes without labels to still have tooltips thus permitting denser graphs. The substring '\N' is substituted in the same manner as for the node label attribute.

## EDGE ATTRIBUTES

**minlen**=*n* where *n* is an integer factor that applies to the edge length (ranks for normal edges, or minimum node separation for flat edges).

**weight**=*n* where *n* is the integer cost of the edge. Values greater than 1 tend to shorten the edge. Weight 0 flat edges are ignored for ordering nodes.

**label**=*text* where *text* may include escaped newlines \n, \l, or \r for centered, left, or right justified lines. If the substring '\T' is found in a label it will be replaced by the tail\_node name. If the substring '\H' is found in a label it will be replaced by the head\_node name. If the substring '\E' value is found in a label it will be replaced by: tail\_node\_name->head\_node\_name or by: tail\_node\_name--head\_node\_name for undirected graphs.

**fontsize**=*n* sets the label type size to *n* points.

**fontname**=*name* sets the label font family name.

**fontcolor**=*colorvalue* sets the label text color.

**style**=**solid dashed dotted bold invis**

**color**=*colorvalue* sets the line color for edges.

**color**=*colorvaluelist* a ':' separated list of *colorvalue* creates parallel edges, one edge for each color.

**dir**=**forward back both none** controls arrow direction.

**tailclip,headclip**=**false** disables endpoint shape clipping.

**href**="url" sets the url for the node in imagemap, PostScript and SVG files. The substrings '\T', '\H', and '\E' are substituted in the same manner as for the edge label attribute.

**URL**="url" ("URL" is a synonym for "href".)

**target**="target" is a target string for client-side imagemaps and SVG, effective when edges have a URL. If the target string is empty, the default, then no target attribute is included in the output. The substrings '\T', '\H', and '\E' are substituted in the same manner as for the edge label attribute.

**tooltip**="tooltip" is a tooltip string for client-side imagemaps effective when edges have a URL. The tooltip string defaults to be the same as the edge label string. The substrings '\T', '\H', and '\E' are substituted in the same manner as for the edge label attribute.

**arrowhead,arrowtail**=**none, normal, inv, dot, odot, invdot, invodot, tee, empty, invempty, open,**

**halfopen, diamond, odiamond, box, obox, crow.**

**arrowsize** (norm\_length=10,norm\_width=5, inv\_length=6,inv\_width=7,dot\_radius=2)

**headlabel,taillabel=string** for port labels. **labelfontcolor,labelfontname,labelfontsize** for head and tail labels. The substrings '\T', '\H', and '\E' are substituted in the same manner as for the edge label attribute.

**headhref="url"** sets the url for the head port in imagemap, PostScript and SVG files. The substrings '\T', '\H', and '\E' are substituted in the same manner as for the edge label attribute.

**headURL="url"** ("headURL" is a synonym for "headhref".)

**headtarget="headtarget"** is a target string for client-side imagemaps and SVG, effective when edge heads have a URL. The headtarget string is used to determine which window of the browser is used for the URL. If the headtarget string is empty, the default, then headtarget defaults to the same value as target for the edge. The substrings '\T', '\H', and '\E' are substituted in the same manner as for the edge label attribute.

**headtooltip="tooltip"** is a tooltip string for client-side imagemaps effective when head ports have a URL. The tooltip string defaults to be the same as the headlabel string. The substrings '\T', '\H', and '\E' are substituted in the same manner as for the edge label attribute.

**tailhref="url"** sets the url for the tail port in imagemap, PostScript and SVG files. The substrings '\T', '\H', and '\E' are substituted in the same manner as for the edge label attribute.

**tailURL="url"** ("tailURL" is a synonym for "tailhref".)

**tailtarget="tailtarget"** is a target string for client-side imagemaps and SVG, effective when edge tails have a URL. The tailtarget string is used to determine which window of the browser is used for the URL. If the tailtarget string is empty, the default, then tailtarget defaults to the same value as target for the edge. The substrings '\T', '\H', and '\E' are substituted in the same manner as for the edge label attribute.

**tailtooltip="tooltip"** is a tooltip string for client-side imagemaps effective when tail ports have a URL. The tooltip string defaults to be the same as the taillabel string. The substrings '\T', '\H', and '\E' are substituted in the same manner as for the edge label attribute.

**labeldistance** and **port\_label\_distance** set distance; also **labelangle** (in degrees CCW)

**decorate** draws line from edge to label.

**samehead,sametail** aim edges having the same value to the same port, using the average landing point.

**constraint=false** causes an edge to be ignored for rank assignment.

**layer=id** or *id:id* or "all" sets the edgess active layers. The empty string means no layers (invisible).

## OPTIONS

**-G** sets a default graph attribute. **-N** and **-E** set default attributes for nodes and edges. Some examples: **-Gsize="7,8"** or **-Nshape=box** or **-Efontsize=8**.

**-Ifile** loads custom PostScript library files. Usually these define custom shapes or styles. If **-I** is given by itself, the standard library is omitted.

**-Tlang** sets the output language as described above.

## EXAMPLE

```
digraph test123 {
  a -> b -> c;
  a -> { x y };
  b [shape=box];
  c [label="hello\nworld",color=blue,fontsize=24,
    fontname="Palatino-Italic",fontcolor=red,style=filled];
  a -> z [label="hi", weight=100];
  x -> z [label="multi-line\nlabel"];
  edge [style=dashed,color=red];
  b -> x;
  {rank=same; b x }
```

}

## **BUGS**

Edge splines can overlap unintentionally.

Flat edge labels are slightly broken. Intercluster edge labels are totally broken.

## **SEE ALSO**

doty(1)  
neato(1)  
circo(1)  
twopi(1)  
fdp(1)  
tcldot(n)  
xcolors(1)  
libgraph(3)

E. R. Gansner, S. C. North, K. P. Vo, "DAG — A Program to Draw Directed Graphs", *Software — Practice and Experience* 17(1), 1988, pp. 1047-1062.

E. R. Gansner, E. Koutsofios, S. C. North, K. P. Vo, "A Technique for Drawing Directed Graphs," *IEEE Trans. on Soft. Eng.* 19(3), 1993, pp. 214-230.

S. North and E. Koutsofios, "Applications of graph visualization", *Graphics Interface* 94, pp. 234-245.

E. Koutsofios and S. C. North, "Drawing Graphs with dot," Available on [research.att.com](http://research.att.com) in `dist/drawdag/dotguide.ps.Z`.

The GIF driver is a publically available library from the Quest Center at Cold Spring Harbor Labs (courtesy Tom Boutell, [boutell@netcom.com](mailto:boutell@netcom.com)). The Truetype font renderer for the GIF driver was written by the Freetype Project (David Turner, Robert Wilhelm, and Werner Lemberg) (who can be contacted at [freetype-devel@lists.lrz-muenchen.de](mailto:freetype-devel@lists.lrz-muenchen.de)).