NAME

osage – filter for drawing clustered graphs

SYNOPSIS

osage [**–(G|N|E)***name*=*value*] [**–T***lang*] [**–l***libfile*] [**–o***outfile*] [**–O**] [**–P**] [**–v**] [**–V**] [files]

DESCRIPTION

osage draws clustered graphs. As input, it takes any graph in the *dot* format. It draws the graph recursively. At each level, there will be a collection of nodes and a collection of cluster subgraphs. The internals of each cluster subgraph are laid out, then the cluster subgraphs and nodes at the current level are positioned relative to each other, treating each cluster subgraph as a node.

At each level, the nodes and cluster subgraphs are viewed as rectangles to be packed together. At present, edges are ignored during packing. Packing is done using the standard packing functions. In particular, the graph attributes **pack** and **packmode** control the layout. Each graph and cluster can specify its own values for these attributes. Remember also that a cluster inherits its attribute values from its parent graph.

After all nodes and clusters, edges are routed based on the value of the **splines** attribute.

OUTPUT FORMATS

Osage uses an extensible plugin mechanism for its output renderers, so to see what output formats your installation of osage supports you can use ‘‘osage –Txxx’’ (where xxx is an unlikely format) and check the warning message. Also, The plugin mechanism supports multiple implementations of the output formats. To see what variants are available, use, for example: ‘‘osage –Tpng:’’ and to force a particular variant, use, for example: ‘‘osage –Tpng:gd’’

Traditionally, osage supports the following: **–Tps** (PostScript), **–Tsvg –Tsvgz** (Structured Vector Graphics), **–Tfig** (XFIG graphics), **–Tmif** (FrameMaker graphics), **–Thpgl** (HP pen plotters), and **–Tpcl** (Laserjet printers), **–Tpng –Tgif** (bitmap graphics), **–Timap** (imagemap files for httpd servers for each node or edge that has a non-null "href" attribute.), **–Tcmapx** (client-side imagemap for use in html and xhtml). Additional less common or more special-purpose output formats can be found at https://www.graphviz.org/doc/info/output.html.)

Alternative plugins providing support for a given output format can be found from the error message resulting from appending a ':' to the format. e.g. **–Tpng:** The first plugin listed is always the default.

GRAPH FILE LANGUAGE

Here is a synopsis of the graph file language, normally using the extension **.gv**, for graphs:

[**strict**] (**graph|digraph**) *name* { *statement-list* }
Is the top level graph. If the graph is **strict** then multiple edges are not allowed between the same pairs of nodes. If it is a directed graph, indicated by **digraph**, then the *edgeop* must be "–>". If it is an undirected **graph** then the *edgeop* must be "––". 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 *edgeop* n1 *edgeop ... edgeop* 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**.

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

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

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

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.

splines. If set to *true*, edges are drawn as splines. If set to *polyline*, edges are drawn as polylines. If set to *ortho*, edges are drawn as orthogonal polylines. In all of these cases, the nodes may not overlap. If **splines=***false* or **splines=***line*, edges are drawn as line segments. The default is *false*.

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 oval circle egg triangle box diamond trapezium parallelogram house hexagon octagon note tab box3d component.** (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. The string '\G' value will be replaced by the graph 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 "*#rrggbb" (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 supersedes 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 substrings '\N' and '\G' are substituted in the same manner as for the node label attribute. Additionally the substring '\L' is substituted with the node label string.

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 substrings '\N' and '\G' are substituted in the same manner as for the node label attribute. Additionally the substring '\L' is substituted with the node label string.

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 substrings '\N' and '\G' are substituted in the same manner as for the node label attribute. Additionally the substring '\L' is substituted with the node label string.

EDGE ATTRIBUTES

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 If the substring '\G' is found in a label it will be replaced by the graph 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', '\E' and '\G' are substituted in the same manner as for the edge label attribute. Additionally the substring '\L' is substituted with the edge label string.

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', '\E' and '\G' are substituted in the same manner as for the edge label attribute. Additionally the substring '\L' is substituted with the edge label string.

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', '\E' and '\G' are substituted in the same manner as for the edge label attribute. Additionally the substring '\L' is substituted with the edge label string.

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', '\E' and '\G' are substituted in the same manner as for the edge label attribute. Additionally the substring '\L' is substituted with the edge label string.

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

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', '\E' and '\G' are substituted in the same manner as for the edge label attribute. Additionally the substring '\L' is substituted with the edge label string.

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. Additionally the substring '\L' is substituted with the edge label string.

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

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', '\E' and '\G' are substituted in the same manner as for the edge label attribute. Additionally the substring '\L' is substituted with the edge label string.

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', '\E' and '\G' are substituted in the same manner as for the edge label attribute. Additionally the substring '\L' is substituted

with the edge label string.

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.

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

(neato-specific attributes)
w=*f* sets the weight (spring constant) of an edge to the given floating point value. The default is 1.0; greater values make the edge tend more toward its optimal length.

len=*f* sets the optimal length of an edge. The default is 1.0.

COMMAND LINE OPTIONS

−G sets a default graph attribute.
−N sets a default node attribute.
−E sets a default edge attribute. Example: **−Gsize="7,8" −Nshape=box −Efontsize=8**

−l*file* loads custom PostScript library files. Usually these define custom shapes or styles. If **−l** is given by itself, the standard library is omitted.

−T*lang* sets the output language as described above.

−O automatically generate output filenames based on the input filename and the −T format.

−o*file* write output to *file*.

−x reduce graph.

−Lg don't use grid.

−LO use old attractive force.

−Ln*i* set number of iterations to *i*.

−LU*i* set unscaled factor to *i*.

−LC*v* set overlap expansion factor to *v*.

−LT[*]*v* set temperature (temperature factor) to *v*.

−v (verbose) prints various information useful for debugging.

−V (version) prints version information and exits.

−? prints the usage and exits.

EXAMPLES

```
digraph test123 {
    pack=8
    subgraph cluster0 {
      packmode=array
      x y x0 y0 x1
      subgraph cluster1 {
        m n
      }
    }
    b [shape=box];
    c [label="hello\nworld",color=blue,fontsize=24,
        fontname="Palatino−Italic",fontcolor=red,style=filled];
    a −> z
    x −> z
    a −> b −> c;
    a −> {x y};
```

```
    edge [style=dashed,color=red];
    b –> x;
}
```

AUTHORS

Emden R. Gansner <erg@research.att.com>

SEE ALSO

This man page contains only a small amount of the information related to the Graphviz layout programs. The most complete information can be found at https://www.graphviz.org/documentation/, especially in the on-line reference pages. Most of these documents are also available in the *doc* and *doc/info* subtrees in the source and binary distributions.

dot(1)