

NAME

gvmap – find clusters and create a geographical map highlighting clusters.

SYNOPSIS

gvmap [**-ekv?**] [*options*] [**-o** *outfile*] [*files*]

DESCRIPTION

gvmap takes as input a graph in DOT format, finds node clusters and produces a rendering of the graph as a geographic-style map, with clusters highlighted, in xdot format.

The input graph must have node positions and width/height information defined, and nodes must not overlap.

OPTIONS

The following options are supported:

- a k** The integer k specifies the average number of artificial points added along the bounding box of the labels. Such artificial points are added to avoid a country boundary cutting through the boundary box of the labels. Computing time is proportional to k; hence, for large graphs, a small value of k is suggested. If k = -1, a suitable value of k is automatically selected based on the graph size. By default k = -1.
- b v** The real number v specifies the line width used to draw the polygon boundaries, with v < 0 for no line. By default v = 0.
- c k** The integer k specifies color scheme used to color the countries. By default k = 1.

Acceptable values are:

- 0 : no polygons
- 1 : pastel
- 2 : blue to yellow
- 3 : white to red
- 4 : light grey to red
- 5 : primary colors
- 6 : sequential single hue red
- 7 : sequential single hue lighter red
- 8 : light grey

- C d** The integer d specifies the maximum number of clusters (countries) allowed. By default d = 0, which means that there is no limit.
- d d** The integer d specifies the random seed used during color assignment optimization that maximize color difference between neighboring countries.
- e** If specified, edges will be included in the final output.
- g c** Specifies the bounding box color. If not specified, a bounding box is not drawn.
- k** If specified, increases the randomness of outer boundary.
- r k** The number of random points k (integer) used to define sea and lake boundaries. If 0, auto assigned. By default v = 0
- s v** The real number v specifies the depth of the sea and lake shores in points. If 0, auto assigned. By default v = 0.
- O** Do NOT do color assignment optimization that maximizes color difference between neighboring countries
- o <file>**
Put output in <file>. Default output is stdout
- v** Verbose mode.
- z c** Specified the polygon line color. Default is black.

EXAMPLES

Given a graph `foo.gv`, one way to generate a layout and highlight the clusters is to first select a layout engine with a suitable overlap removal method, then feed the output to `gvmap`, and finally render the map using specific graphics format. For example, the following pipeline creates a map with edges in semi-transparent light gray and nodes laid out using `sfdp`:

```
sfdp -Goverlap=prism foo.gv | gvmap -e | neato -n2 -Ecolor=#55555522 -Tpng > foo.png
```

The shell script `gvmap.sh` provides a shorthand for such pipelines. For example, the above pipeline can be achieved using

```
gvmap.sh -Ae -Ecolor=#55555522 -Tpng foo.gv > foo.png
```

AUTHOR

Yifan Hu <yifanhu@research.att.com>

SEE ALSO

`gvmap.sh(1)`, `sfdp(1)`, `neato(1)`, `gvpr(1)`

E. R. Gansner, Y. Hu, S. G. Kobourov, "GMap: Visualizing graphs and clusters as maps," *Proc. Pacific Vis.* 2010, pp. 201-208.