## NAME

gvcolor – flow colors through a ranked digraph

## SYNOPSIS

**gvcolor** [ *files* ]

## USAGE

dot file.gv | gvcolor | dot –T<*format*>

## DESCRIPTION

**gvcolor** (previously known as **colorize)** is a filter that sets node colors from initial seed values. Colors flow along edges from tail to head, and are averaged (as HSB vectors) at nodes. *The graph must already have been processed by dot.* Appropriate choice of initial colors yields drawings in which node colors help to emphasize logical relationships between nodes, even when they are spread far apart in the layout.

Initial colors must be set externally, using the **color** attribute of a node. It is often effective to assign colors to a few key source or sink nodes, manually setting their colors by editing the graph file. Color names are as in *dot(1)*: symbolic names or RGB triples. It is best to choose some easily-distinguished but related colors; not necessarily spaced evenly around the color wheel. For example, blue_green, green, and light_yellow looks better than red, green, blue.

Certain graph attributes control the *gvcolor* algorithm. **flow=back** reverses the flow of colors from heads to tails. **saturation=.1,.9** (or any two numbers between 0 and 1) adjusts the color saturation linearly from least to greatest rank. If **Defcolor** is set, this color value is applied to any node not otherwise colored.

## EXIT STATUS

The following exit values are returned:

**0** Successful completion.

**1** If nodes of the graph do not possess a ‘‘pos’’ attribute.

## BUGS

It would be nice to make the program work without relying on an initial pass through **dot**.

## AUTHORS

Stephen C. North <north@research.att.com>
Emden R. Gansner <erg@research.att.com>

## SEE ALSO

gc(1), dot(1), gvpr(1), ccomps(1), sccmap(1), tred(1), libgraph(3)