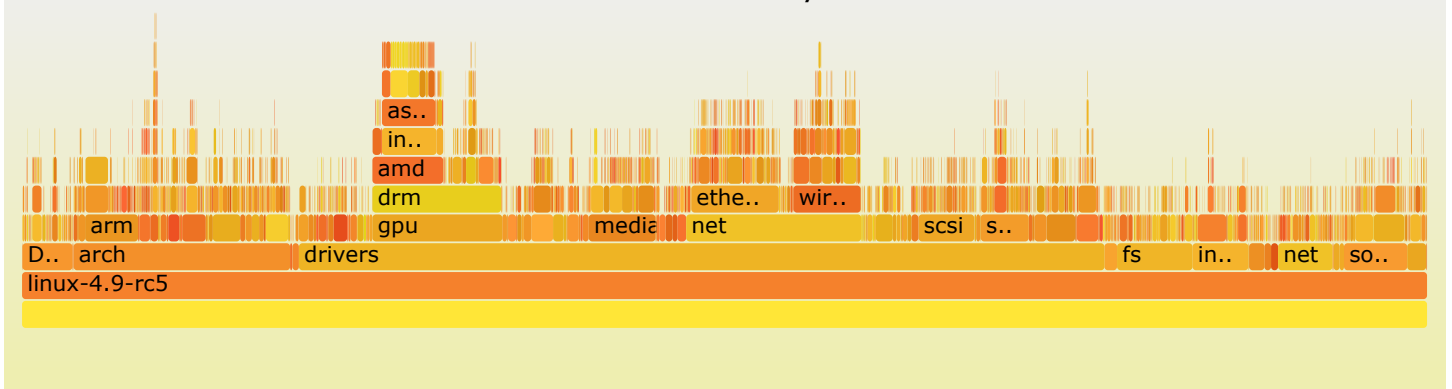


Where has my disk space gone? Flame graphs for file systems

05 Feb 2017

My laptop was recently running low on available disk space, and it was a mystery as to why. I have different tools to explore the file system, including running the "find / -ls" command from a terminal, but they can be time consuming to use. I wanted a big picture view of space by directories, subdirectories, and so on.

I've created a simple open source tool to do this, using flame graphs as the final visualization. To demonstrate, here's the space consumed by the Linux 4.9-rc5 source code. Click to zoom, and Ctrl-F to search ([SVG](#)):



If you are new to flame graphs, see my [flame graphs](#) page. In this case, width corresponds to total size. I created them for visualizing stack traces, but since they are a generic hierarchical visualization (technically an [adjacency diagram with an inverted icicle layout](#)), they are suited for the hierarchy of directories as well.

I've also used this to diagnose a similar problem with a friend's laptop, which turned out to be due to a backup application consuming space in a directory completely unknown to them.

The following sections show how to create one yourself. Start by opening a terminal session so you can use the command line.

Using git

If you have the "git" command, you can fetch the [FlameGraph](#) repository and run the commands from it:

```
git clone https://github.com/brendangregg/FlameGraph
cd FlameGraph
./files.pl /Users | ./flamegraph.pl --hash --countname=bytes > out.svg
```

Then open out.svg in a browser. Change "/Users" to be the directory you want to visualize. This could be "/" for everything (provided you don't have removable storage or network file systems mounted, which if you do, it would include them in the report by accident).

Without git

If you don't have git, you can download the two Perl programs straight from github: [files.pl](#) and [flamegraph.pl](#), either using wget or download them via your browser (save as). The steps can then be:

```
chmod 755 files.pl flamegraph.pl
./files.pl /Users | ./flamegraph.pl --hash --countname=bytes > out.svg
```

Again, change "/Users" to be the directory you want to visualize, then open out.svg in a browser.

Linux source example

For reference, the Linux source example I included above was created using:

```
files.pl linux-4.9-rc5 | flamegraph.pl --hash --countname=bytes \
--title="Linux source tree by file size" --width=800 > files_linux49.svg
```

You can customize the flame graph using options:

```
$ ./flamegraph.pl -h
Option h is ambiguous (hash, height, help)
USAGE: ./flamegraph.pl [options] infile > outfile.svg

--title      # change title text
--width      # width of image (default 1200)
--height     # height of each frame (default 16)
--minwidth   # omit smaller functions (default 0.1 pixels)
--fonttype   # font type (default "Verdana")
--fontsize   # font size (default 12)
--countname  # count type label (default "samples")
--nametype   # name type label (default "Function:")
--colors     # set color palette. choices are: hot (default), mem, io,
              # wakeup, chain, java, js, perl, red, green, blue, aqua,
              # yellow, purple, orange
--hash       # colors are keyed by function name hash
--cp         # use consistent palette (palette.map)
--reverse    # generate stack-reversed flame graph
--inverted   # icicle graph
--negate     # switch differential hues (blue<->red)
--help      # this message

eg,
./flamegraph.pl --title="Flame Graph: malloc()" trace.txt > graph.svg
```

You might need to know about the `--minwidth` option: rectangles thinner than this (1/10th of a pixel when zoomed out) will be elided, to conserve space in the SVG. But that can mean things are missing when you zoom in. If it's a problem, you can set `minwidth` to 0.

Update: see my follow-on post [Flame Graphs vs Tree Maps vs Sunburst](#).

Comments for this thread are now closed



18 Comments

Brendan Gregg's Blog

Login ▾

Recommend 4

Tweet

Share

Sort by Best ▾



Paul Ivanov • 2 years ago

For anyone looking to get similar information as text directly on the command line, I've been happily using [duviz](#) from Stefaan Lippens for years.



71 ^ | ▾ • Share ▾



brendangregg Mod ➔ Paul Ivanov • 2 years ago

Nice! That is actually an icicle plot adjacency diagram, just like flame graphs. An ASCII flame graph.

Interesting that they sort from subdirectory size on the y axis. I sort alphabetically (which was

interesting that they sort from subdirectory size on the x-axis. I sort alphabetically (which was originally necessary for merging stack samples).

1 ^ | v • Share ›



Justin Derleth • 2 years ago

Just an FYI on macOS Sierra 10.12.3 I am getting the following output (presumably 5417 times).

```
sudo ./files.pl / | sudo ./flamegraph.pl --hash --countname=bytes > out.svg
```

Use of uninitialized value \$size in concatenation (.) or string at ./files.pl line 39.

Ignored 5417 lines with invalid format

It did however generate a SVG that appears to be correct or within reason.

1 ^ | v • Share ›



brendangregg Mod ➔ Justin Derleth • 2 years ago

Ok, thanks, fixed (just pushed to github). I think those were /dev files.

^ | v • Share ›



Justin Derleth ➔ brendangregg • 2 years ago

I can confirm its fixed. Thanks

^ | v • Share ›



Jonathan Briggs • 2 years ago

Flame graphs are neat.

In case you hadn't seen them there are other tools for this. On Windows, windirstat. On Gnome, baobab.

1 ^ | v • Share ›



Mosè Bottacini ➔ Jonathan Briggs • 2 years ago

on Windows, i like http://www.uderzo.it/main_p... more

^ | v • Share ›



brendangregg Mod ➔ Jonathan Briggs • 2 years ago

I'm surprised those tools don't use the flame graph layout (which is just an adjacency diagram with an inverted icicle layout), which works better than sunburst, treemap, tree lists, pie charts, etc.

^ | v • Share ›



Sigi Kiermayer ➔ brendangregg • 2 years ago

I favour tree maps. It makes it easier for me to see the big chunks.

At least out of my experience what i'm looking for when i try to cleanup my hddrive.

Otherwise i just use `du -hs * | sort -h`

^ | v • Share ›



Ryan Aslett ➔ brendangregg • 2 years ago

I strongly favor cushioned tree maps for visualizing hierarchical aggregations.. `kdirstat` for linux and `diskinventoryX` for osx.. (following this paper on cushioned tree maps: <https://www.win.tue.nl/~van...> - there are some 'fake' cushioned maps that dont use the algorithm - they just spew out some gradient that doesn't help the visualization, but those two disk utilities are great example implementations.

I've often thought that cushioned tree maps could actually provide a better visualization than flame graphs for revealing hotspots in a hierarchy of data.

^ | v • Share ›



brendangregg Mod → Ryan Aslett • 2 years ago

I don't favor tree maps, as comparing box sizes vs line lengths is a more difficult perceptive task. Although I do like how they make use of all available space.

I think if I were making a serious product to visualize file system usage, I would include flame graphs, tree maps, and sunburst, all as optional ways to visualize the same data.

^ | v • Share ›



DaisyDisk → brendangregg • 2 years ago

We actually tried this kind of graphs (flame graph) in early DaisyDisk prototypes, as well as other visualizations too, but we ended up using the sunburst because it performed better (UX-wise) in our testing.

The upside of the flame graph is that, unlike the sunburst, it retains the relation between segment areas (more precisely, bar lengths) across different layers. However it's rather a minor benefit because such comparisons are rare.

The upside of the sunburst is that it gives a better perception of relative size – you can easily "weigh" a share of the pie, but it's kind of difficult to compare lengths of horizontal bars relative to the total length of the whole graph.

Other than that, sunburst graph and flame graph are mathematically equivalent, just in different coordinates – Polar vs. Cartesian.

Tree maps have their upsides too. They maintain relations of areas in proportion to file sizes, and their biggest benefit is that you immediately see the ultimate space hogs without drilling down through folders first. And their biggest downside is that they don't immediately (visually) provide information about parent folders. The location of a file is an important information to understand and make the decision to delete it. The files are grouped in blocks by folder which are located in random order, which also makes navigation more difficult. Also tree maps do not allow progressive disclosure of information and are kind of overwhelming with the amount of detail they display upfront. That said, with properly honed implementation tree maps might work too.

3 ^ | v • Share ›



Mpekris Mezes → brendangregg • 2 years ago

I would argue that treemap makes the best use of available estate in a rectangular setting. Especially if it allows you to zoom in one of the tile for better scaling. Flamegraph is a cool experiment but you basically lose/underutilize a dimension in the use case of file sizing and similar.

^ | v • Share ›



brendangregg Mod → Mpekris Mezes • 2 years ago

Yes, tree maps make good use of space. However, comparing box sizes vs lengths is a more difficult perceptive task. They can also be harder to label.

^ | v • Share ›



Petah → brendangregg • 2 years ago

Or Treesize <https://www.jam-software.co...>

^ | v • Share ›



twoflower234 • 2 years ago

Very neat, Thanks!

^ | v • Share ›



LarryCohen2014 • 2 years ago

Brendan, slightly off topic... could you please fix your fonts on this page? Your choice of windows-only fonts looks atrocious under Linux and non-MS OSes. Thanks.

^ | v • Share ›



brendangregg Mod → LarryCohen2014 • 2 years ago

Ok, I know I need to fix it on mobile too....

1 ^ | v • Share ›

Subscribe **Add Disqus to your site** **Add DisqusAdd** **Disqus' Privacy Policy** **Privacy PolicyPrivacy**

You can comment here, but I can't guarantee your comment will remain here forever: I might switch comment systems at some point (eg, if disqus add advertisements).

Copyright 2017 Brendan Gregg.
[About this blog](#)