FlameGraphs 201

So Brendan. Very useful. Much perf. Wow.

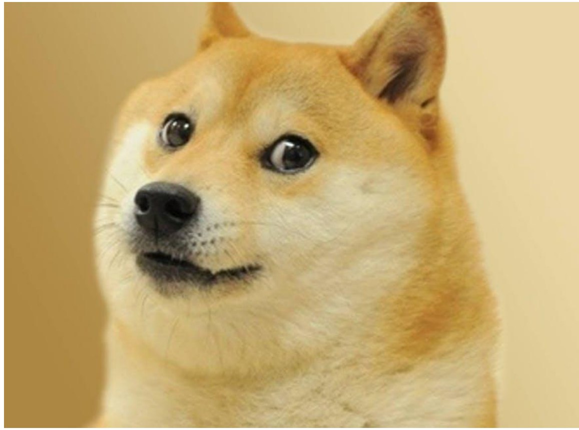
Marcos Albe

August 2019

PERCONA

Agenda

- So Brendan



- Much perf

- Very Useful

- Wow

So Brendan

Or "What are FlameGraphs?"

What are FlameGraphs

Flame graphs are a visualization of profiled software, allowing the most frequent code-paths to be identified quickly and accurately

~ Brendan Gregg

What are FlameGraphs: code path

```
int main(){
    ...
    GetBookList(book);
    ...
}

void GetBookList(Book book){
    …
    while (book =
    GetBook(book);
    ...
}
```

```
void GetBook(Book book){
   a = GetAuthor(book);
   pub = GetPublisher(book);
   p = GetPrints(book).toString();

   printf("%s published by %s [prints: %s]",
a,pub,p);
    ...
}
```

What are FlameGraphs: code path

Possible call chains

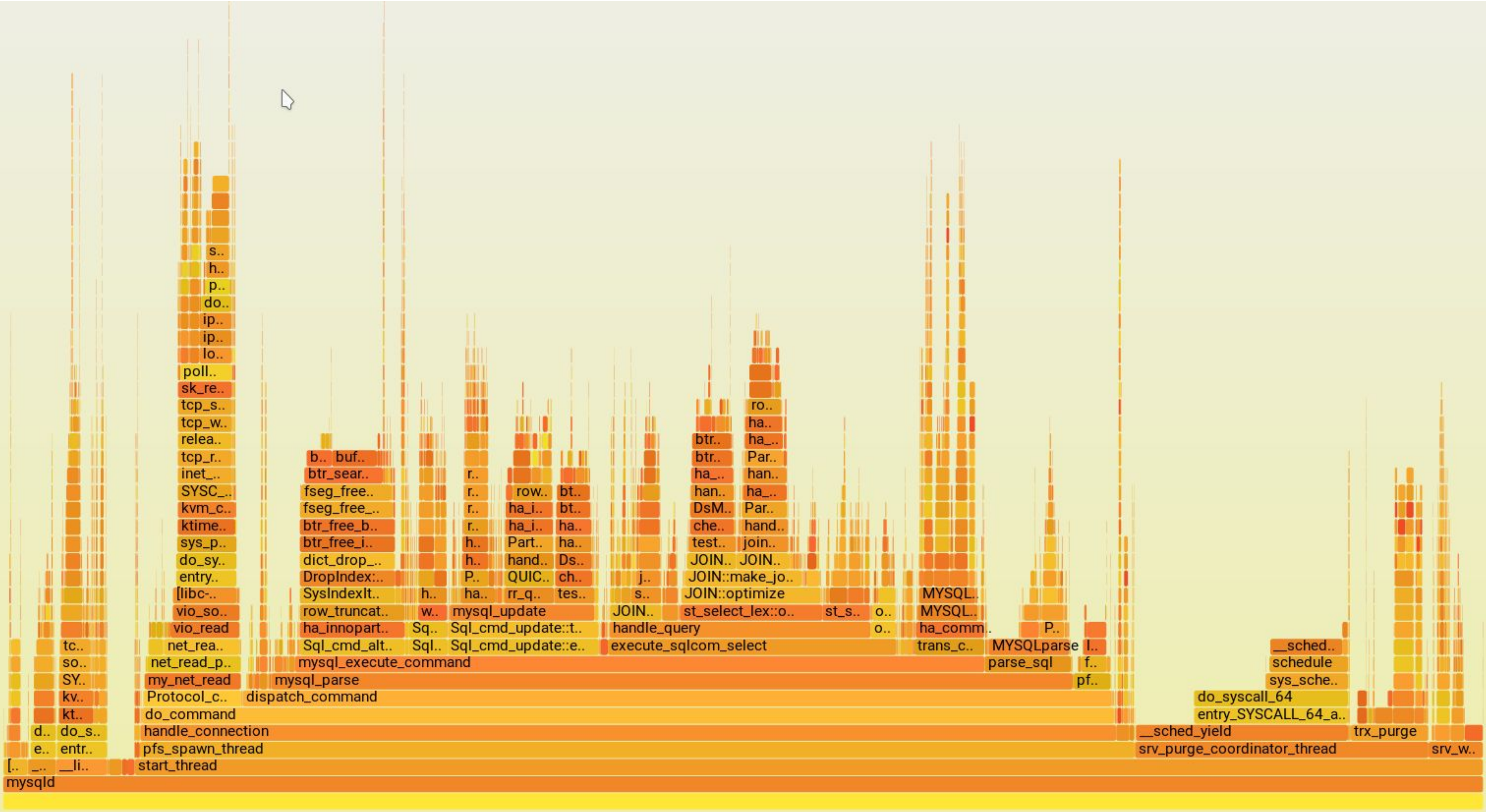
main() -> GetBooksList() -> GetBook() -> GetAuthor()...

main() -> GetBooksList() -> GetBook() -> GetEditorial()...

main() -> GetBooksList() -> GetBook() -> GetPrints()...

What are FlameGraphs: profiling

Captures of metrics of some type for later aggregation.


s..
h..
p..
do..
ip..
ip..
lo..
poll..
sk_re..
tcp_s..
tcp_w..
relea..
tcp_r..
inet_..
SYSC_..
kvm_c..
ktime..
sys_p..
do_sy..
entry..
[libc-..
vio_so..
vio_read
net_rea..
net_read_p..
my_net_read
Protocol_c..
do_command
handle_connection
pfs_spawn_thread
start_thread
mysqld
tc..
so..
SY..
kv..
kt..
d..
do_s..
e..
entr..
[..
_..
__li..
ro..
ha..
ha_..
Par..
han..
ha_..
Par..
hand..
join..
JOIN..
b..
buf..
btr_sear..
fseg_free..
fseg_free_..
btr_free_b..
btr_free_i..
dict_drop_..
DropIndex:..
SysIndexIt..
row_truncat..
ha_innopart..
Sql_cmd_alt..
r..
r..
r..
r..
h..
h..
P..
ha..
row..
ha_i..
ha_i..
Part..
hand..
QUIC..
rr_q..
bt..
bt..
ha..
ha..
Ds..
ch..
tes..
h..
w..
mysql_update
Sq..
Sql..
Sql_cmd_update::t..
Sql_cmd_update::e..
mysql_execute_command
mysql_parse
dispatch_command
btr..
btr..
ha_..
han..
DsM..
che..
test..
JOIN..
JOIN::make_jo..
j..
s..
JOIN::optimize
JOIN..
st_select_lex::o..
st_s..
o..
o..
handle_query
execute_sqlcom_select
MYSQL..
MYSQL..
ha_comm..
trans_c..
P..
MYSQLparse
l..
parse_sql
f..
pf..
__sched..
schedule
sys_sche..
do_syscall_64
entry_SYSCALL_64_a..
__sched_yield
trx_purge
srv_purge_coordinator_thread
srv_w..

What are FlameGraphs: demo 101

```
# capture
perf record -a -g -F99 -p $(pgrep -x mysqld) -- sleep 60
# make it machine-readable
perf script > s.out
# fold it
./stackcollapse-perf.pl s.out > s.folded
# graph it
./flamegraph.pl s.folded > s.svg
```

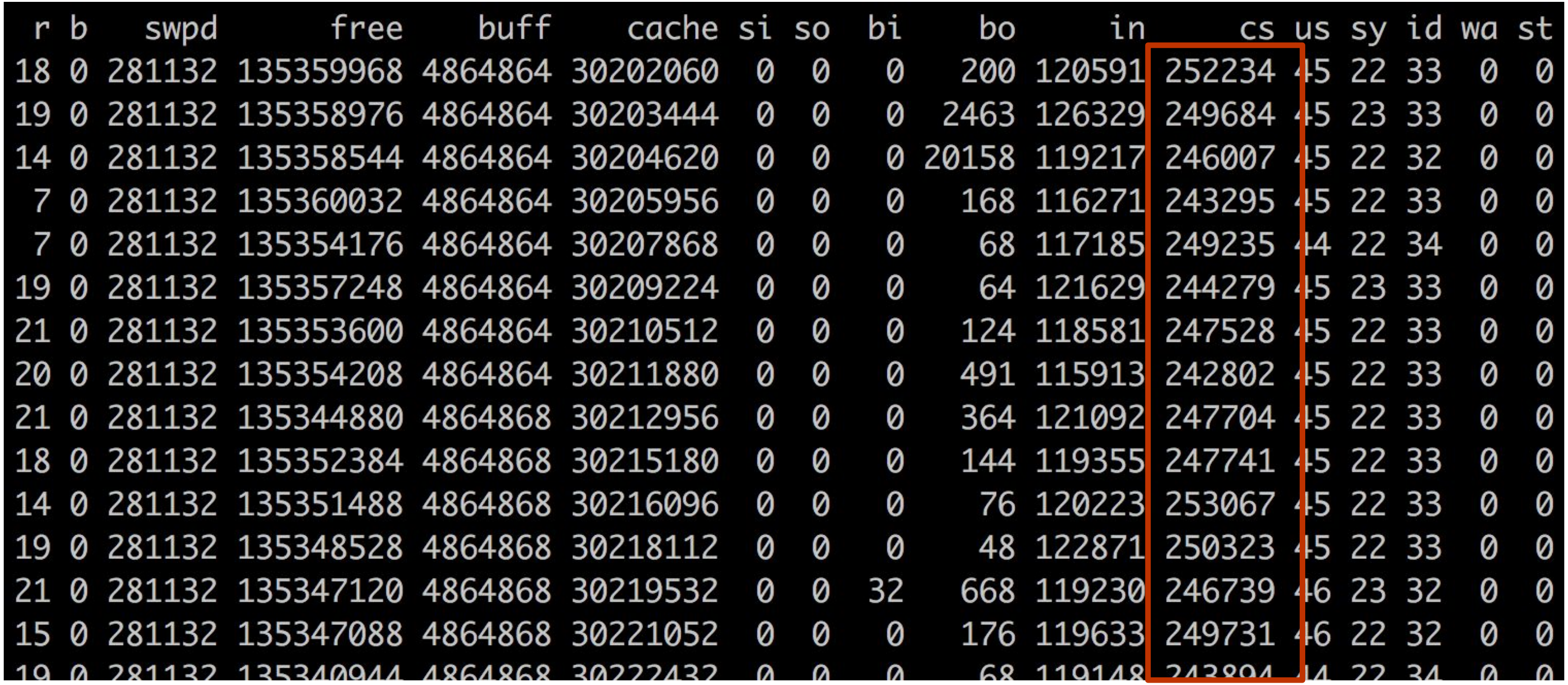
Very useful

Or "Advanced usage"

Advanced usage: workflow

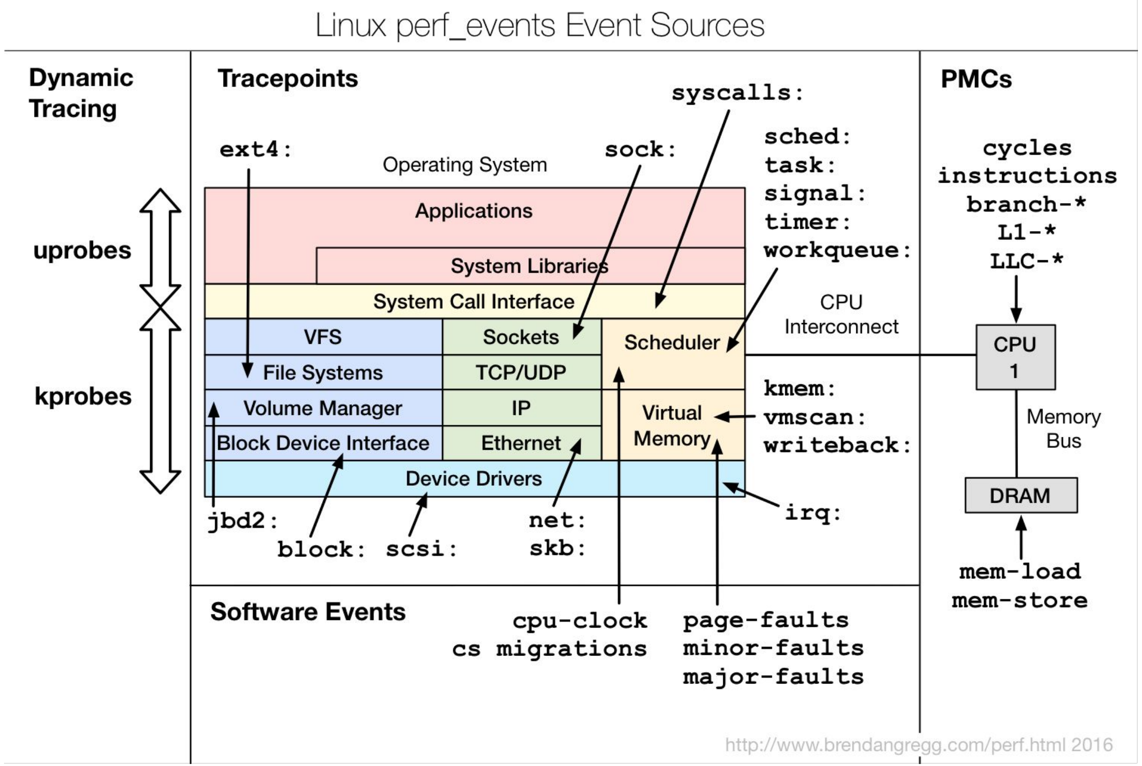
- **Use classic Linux tools to find hot resource**
  - vmstat
  - pidstat
  - mpstat
  - netstat
- **Identify events/probes for the subsystem**
- **Capture filtered perf**
- **Cleanup perf script output (if necessary)**
- **Produce flamegraphs**

Advanced usage: vmstat



```
 r b   swpd      free    buff    cache si so  bi    bo     in     cs us sy id wa st
18 0 281132 135359968 4864864 30202060  0  0   0   200 120591 252234 45 22 33  0  0
19 0 281132 135358976 4864864 30203444  0  0   0  2463 126329 249684 45 23 33  0  0
14 0 281132 135358544 4864864 30204620  0  0   0 20158 119217 246007 45 22 32  0  0
 7 0 281132 135360032 4864864 30205956  0  0   0   168 116271 243295 45 22 33  0  0
 7 0 281132 135354176 4864864 30207868  0  0   0    68 117185 249235 44 22 34  0  0
19 0 281132 135357248 4864864 30209224  0  0   0    64 121629 244279 45 23 33  0  0
21 0 281132 135353600 4864864 30210512  0  0   0   124 118581 247528 45 22 33  0  0
20 0 281132 135354208 4864864 30211880  0  0   0   491 115913 242802 45 22 33  0  0
21 0 281132 135344880 4864868 30212956  0  0   0   364 121092 247704 45 22 33  0  0
18 0 281132 135352384 4864868 30215180  0  0   0   144 119355 247741 45 22 33  0  0
14 0 281132 135351488 4864868 30216096  0  0   0    76 120223 253067 45 22 33  0  0
19 0 281132 135348528 4864868 30218112  0  0   0    48 122871 250323 45 22 33  0  0
21 0 281132 135347120 4864868 30219532  0  0  32   668 119230 246739 46 23 32  0  0
15 0 281132 135347088 4864868 30221052  0  0   0   176 119633 249731 46 22 32  0  0
19 0 281132 135340944 4864868 30222432  0  0   0    68 119148 243894 44 22 34  0  0
```

Advanced usage: event name(s)?

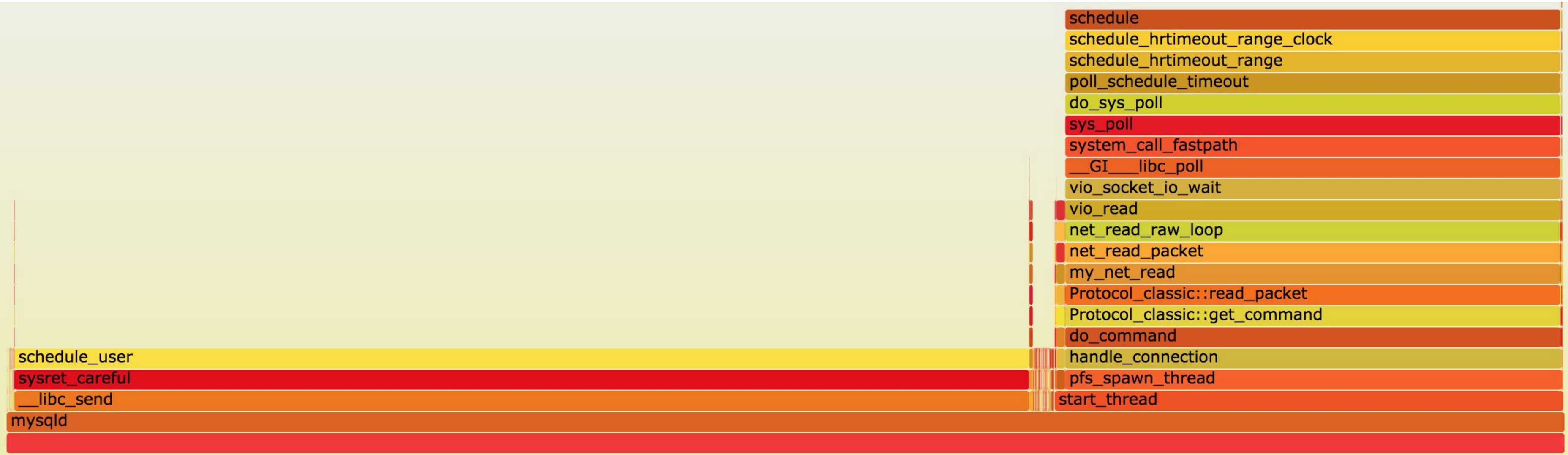


```
sudo perf list
```

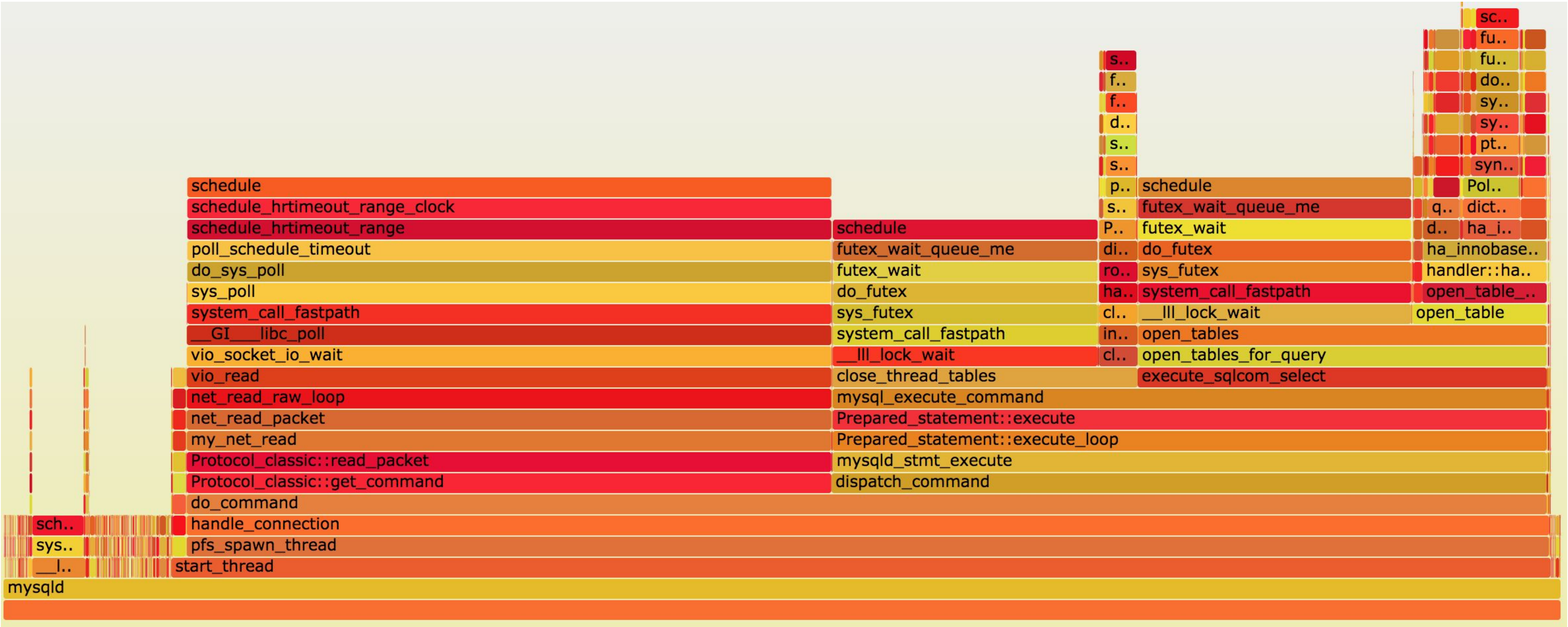
Advanced usage: perf record -e

```
perf record -a -g -F99 -e 'context_switches' -p123 -- sleep 60
```

Advanced usage: baseline w/filter



Advanced usage: problem w/filter

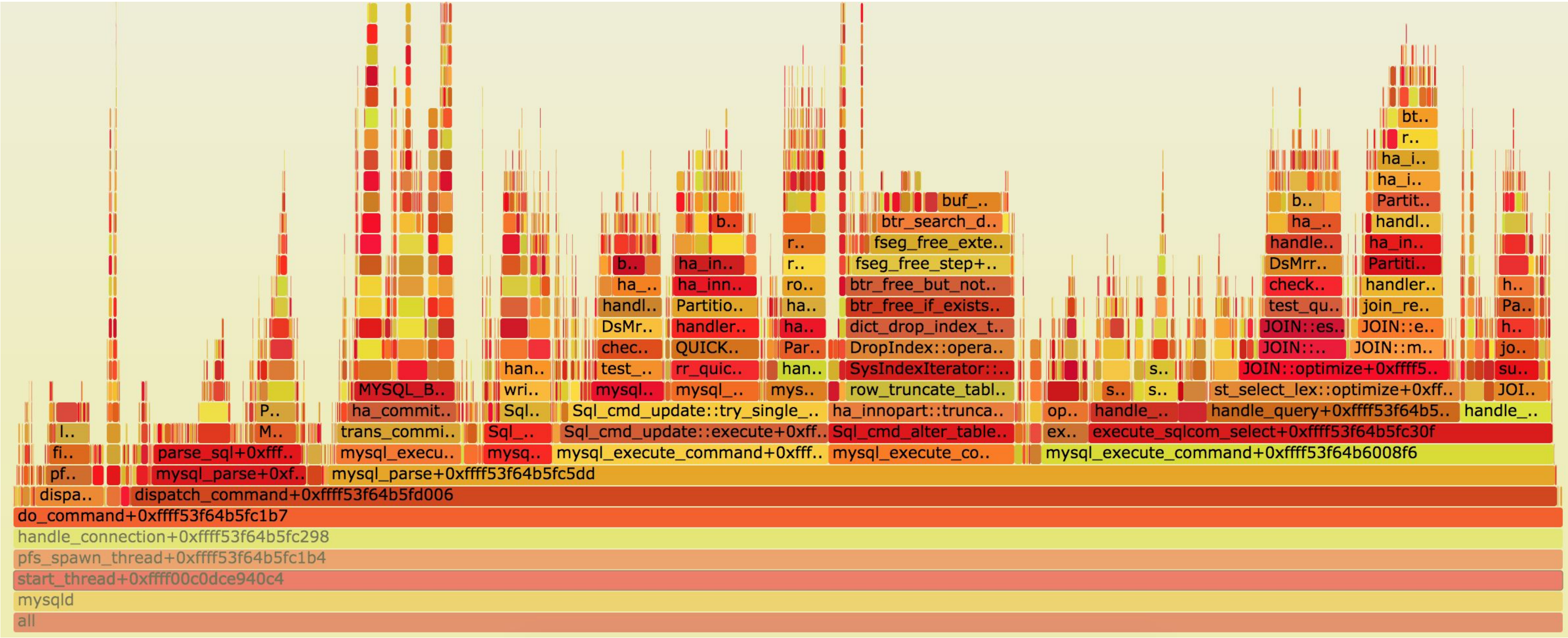


PERCONA

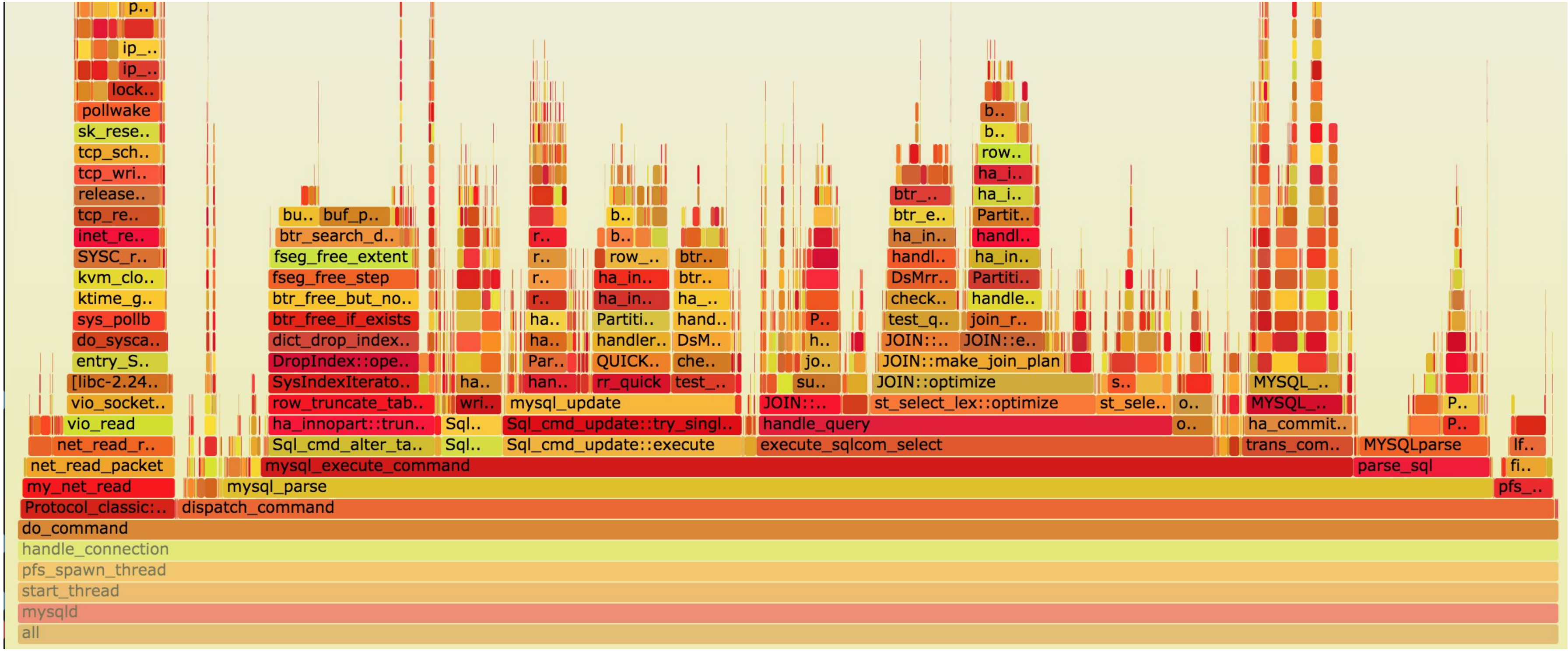
Advanced usage: cleanup

```
perf script | ./stackcollapse-perf.pl > out.perf-folded

grep -v cpu_idle out.perf-folded > nonidle.folded
grep ext4 out.perf-folded > ext4internals.folded
egrep 'system_call.*sys_(read|write)' out.perf-folded > rw.folded

sed -E 's/\+0x[0-9]+//g' < out.perf-folded > nohexaddr.folded
```

Advanced usage: noisy



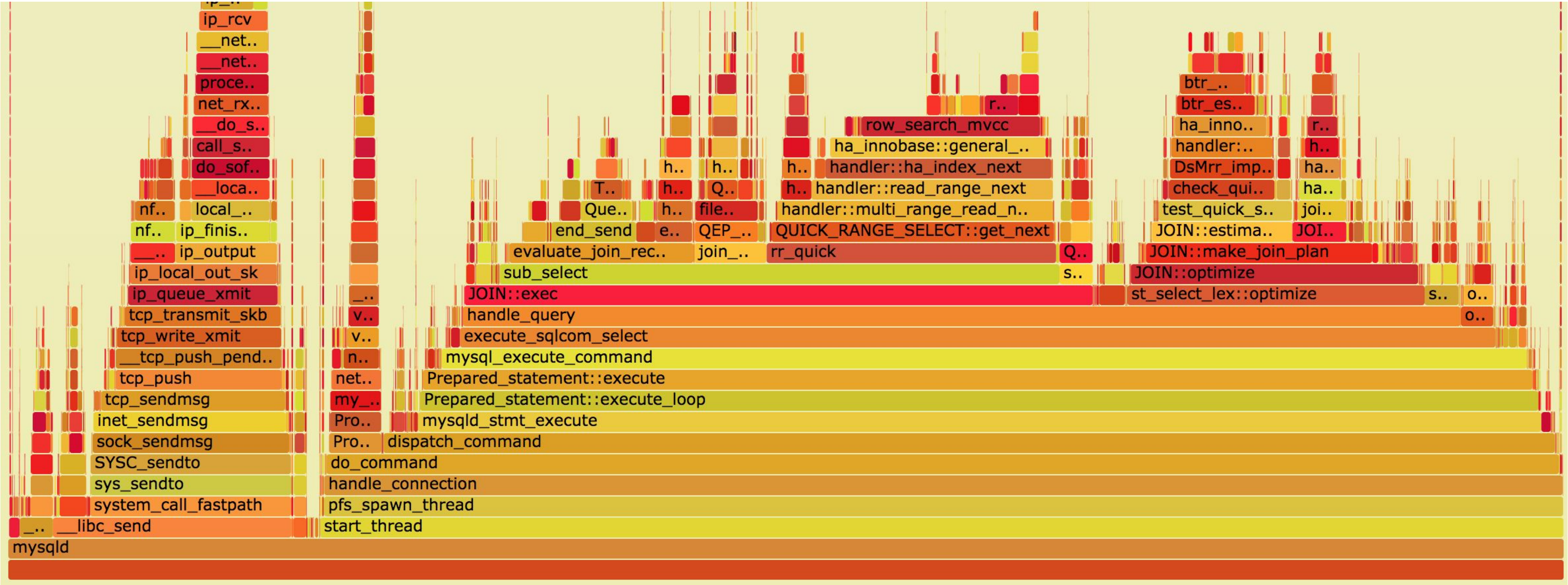
Advanced usage: clean



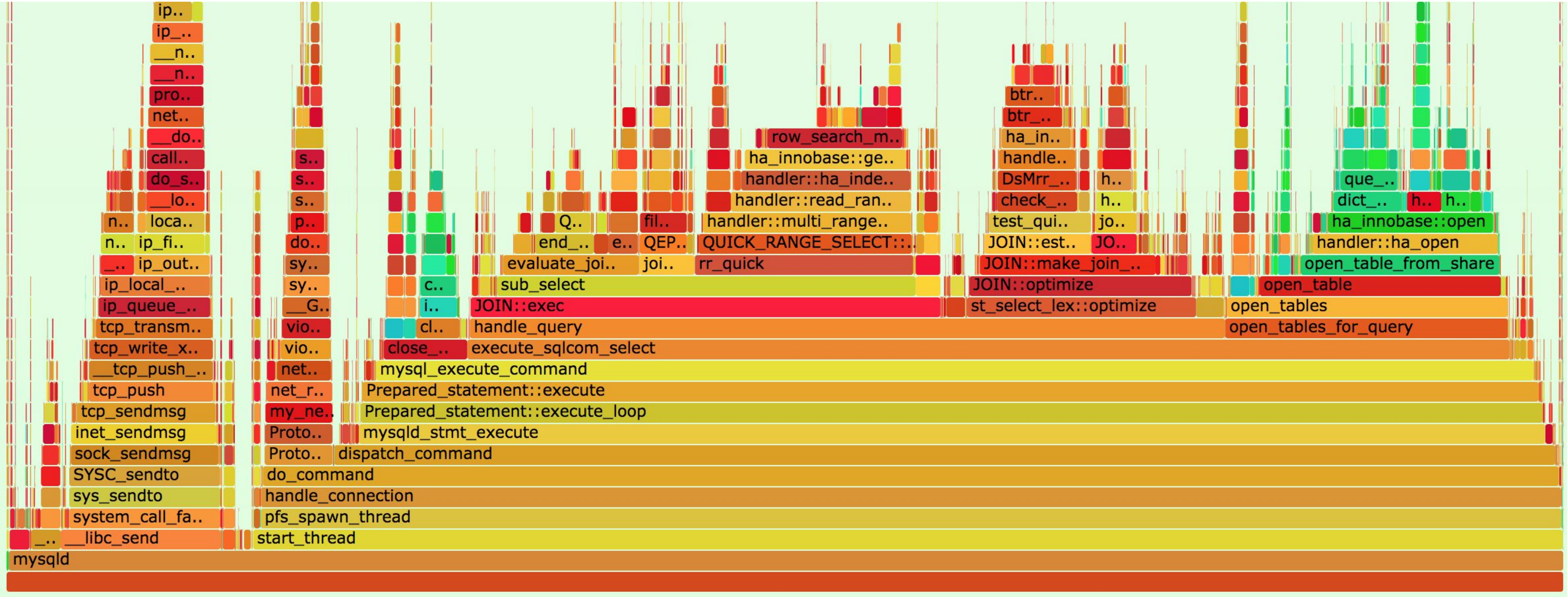
Advanced usage: evil delta graphs

```
# first flamegraph shown was produced with
./flamegraph.pl script.folded > graph.svg

# baseline; record [c]onsistent [p]alette
./flamegraph.pl --cp good.folded > good.svg

# evil delta; use recorded palette + color for delta
./flamegraph.pl --cp --color mem bad.folded > bad.svg
```

Advanced usage: baseline



Advanced usage: evil delta



Much perf

Or "Quick demo"

Wow

Or "Questions?"