Adventures in Ridiculous Benchmarking

.pgpass edition

@gorthx

"What's the maximum number of entries we can have in our .pgpass file?"

"...Why do you ask?"

in #pdxpug...

```
13:30   <@gorthx> | OK so
13:30   <@gorthx> | let me state this another way
13:30   <@gorthx> | ”how big is your .pgpass file”
13:30   <@gorthx> | anybody have one with > 200 entries?
13:59 <@redacted> | OMG I don’t know where this is headed but I want to watch
```

…later that night…

How hard could this be, really

- Create a .pgpass file with garbage entries
- ”Hide” the valid entry at the beginning or end
- Make postgres use this .pgpass file for a very simple query (”SELECT 1;”)
- Time it.

Idle hands are the Devil's playground

- Create the desired size .pgpass file
- Flush memory

```
sync; echo 3 | sudo tee /proc/sys/vm/drop_caches
```

- Run query; time it; flush memory again... repeat several times
- Repeat the above four steps for ever-increasing numbers of garbage rows
- ...and record some system stats at 1-second intervals while I'm doing this

La machine (comm10)

- HP DL-380 G5
- 2 x Quad-Core Xeon 2.0 GHz
- **32GB** Fully Buffered DIMM PC2-5300 8x4GB DR LP Memory
- HP Smart Array P800 Controller
- 8 x 72GB Hot Plug 2.5 SAS 15,000 rpm Hard Drive (Internal)
- Gentoo Base System release 1.12.13* PostgreSQL 9.2.3
- **Not doing anything but this test!**

Hold my beer and watch this.

- Biggest file: 16,581,376 rows, ~675M

Hold my beer and watch this.

- Biggest file: 16,581,376 rows, ~675M
- valid creds at beginning of file:

```
real 0m0.162s
```

Hold my beer and watch this.

- Biggest file: 16,581,376 rows, ~675M
- valid creds at beginning of file:

```
real 0m0.162s
```

- valid creds at end of file:

```
real 0m7.553s
```

Hold my beer and watch this.

- Biggest file: 16,581,376 rows, ~675M
- valid creds at beginning of file:

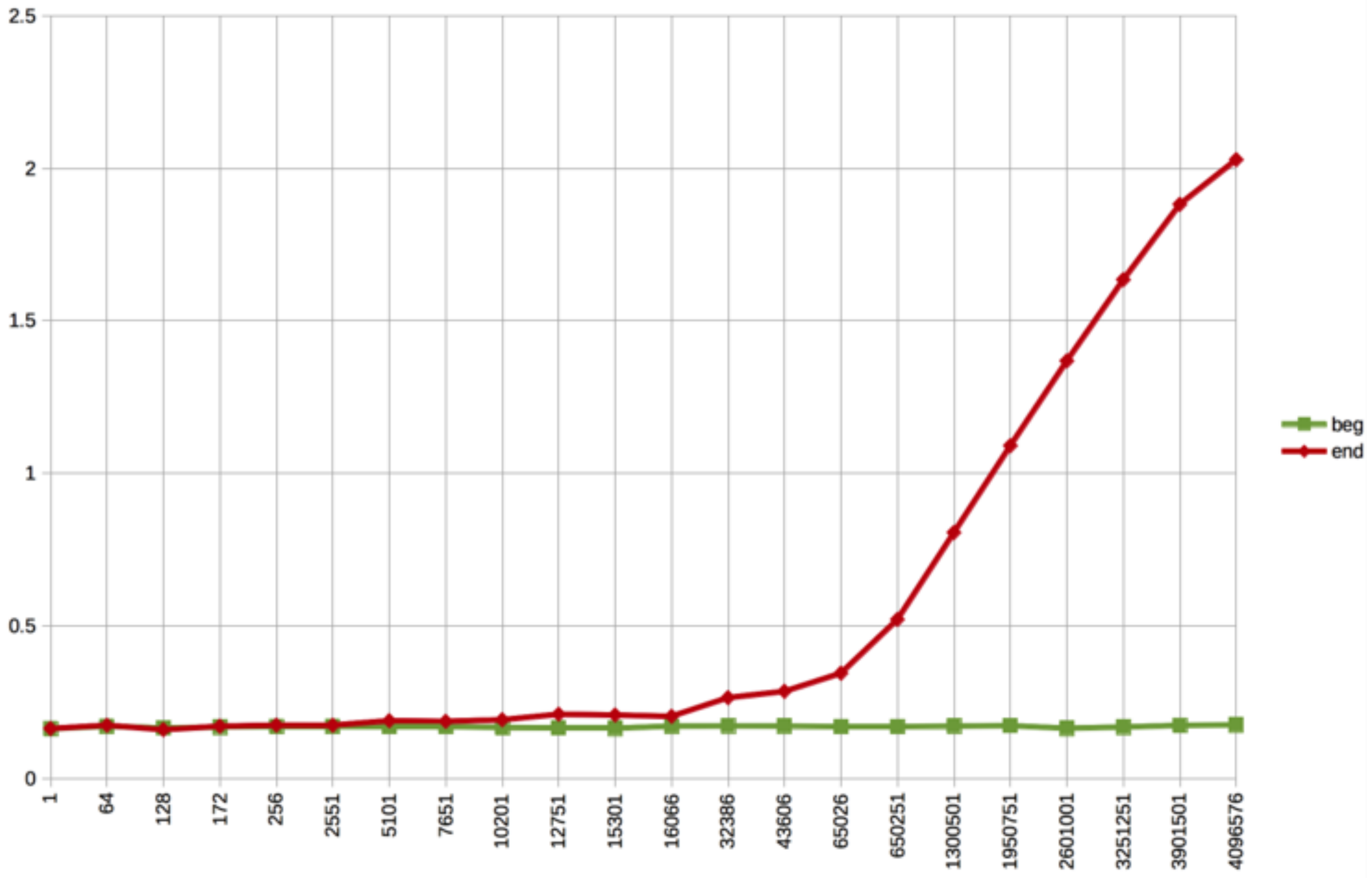
```
real 0m0.162s
```

- valid creds at end of file:

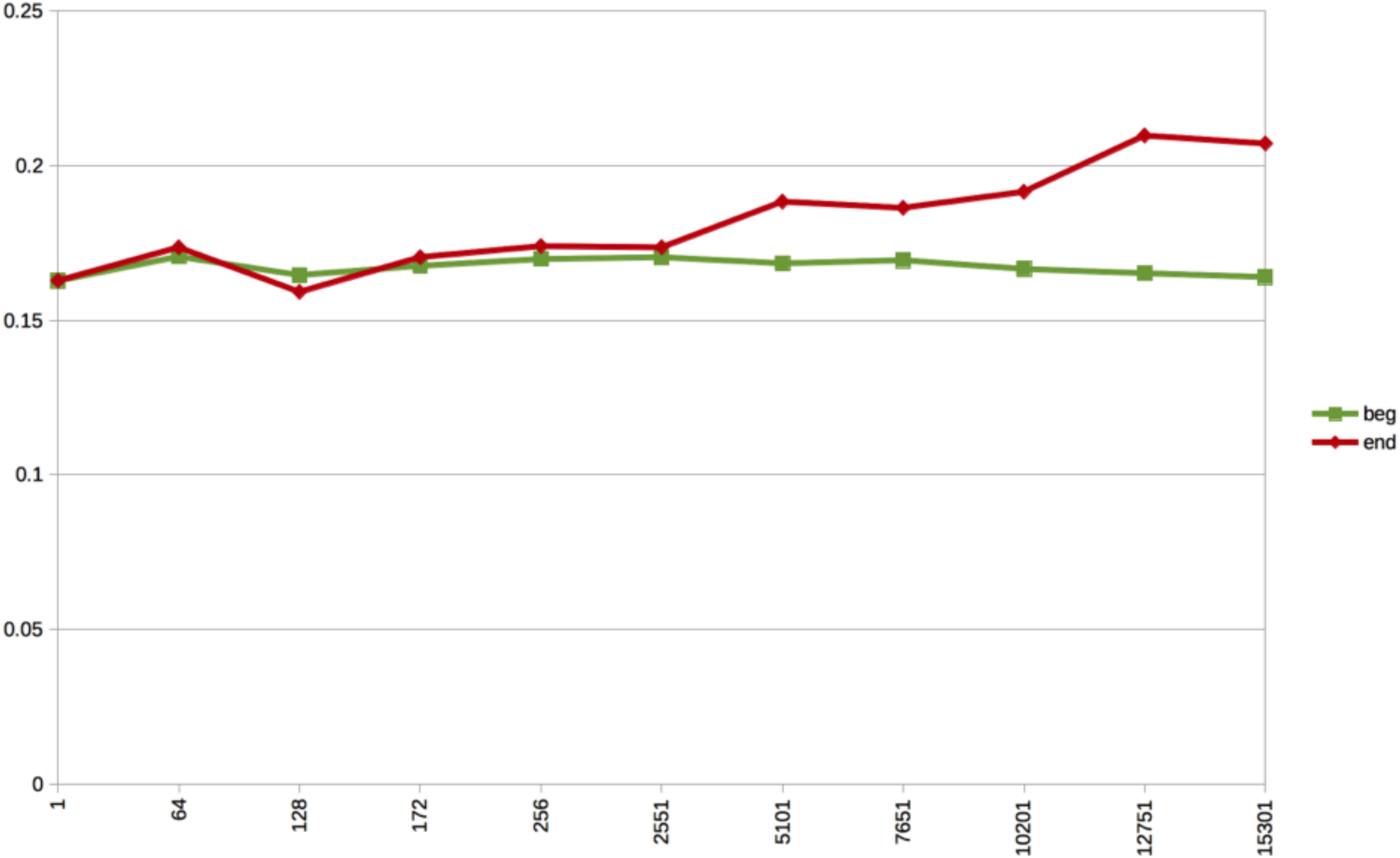
```
real 0m7.553s
```

- CPU went from ~0 to ~10%

Where is the pain point?



Zoom in...


0.25
0.2
0.15
0.1
0.05
0
1
64
128
172
256
2551
5101
7651
10201
12751
15301
beg
end

Just because you can,
doesn't mean you should.