Announcing pg_statviz

Jimmy Angelakos

Senior Solutions Architect

EDB

FOSDEM

2023-02-04

Announcing pg_statviz

- A minimalist **extension and utility pair**
- Time series analysis and visualization of **PostgreSQL** internal statistics

Announcing pg_statviz

- Created for:
  - Snapshotting **cumulative and dynamic** statistics
  - Performing **time series analysis** on them
- Utility can produce visualizations for selected time ranges on the stored stats snapshots

Announcing pg_statviz

- Why?
  - Track PostgreSQL performance over time and potentially perform tuning or troubleshooting

Announcing pg_statviz

- **K.I.S.S.** and **UNIX** philosophies
- Tool aims to be:
  - Modular
  - Minimal
  - Unobtrusive
- Does only what it's meant for: create snapshots of PostgreSQL statistics for visualization and analysis.

Announcing pg_statviz

- Components
  - PostgreSQL extension
  - Python utility for retrieving stored snapshots & creating simple visualizations using **Matplotlib**
- Nothing to put in **shared_preload_libraries**
- No need to restart Postgres

Announcing pg_statviz

- Extension installation:

  CREATE EXTENSION pg_statviz;

- Utility installation:

  $ pip install pg_statviz

Announcing pg_statviz

- Extension can be used by superusers, or any user that has **pg_monitor** role privileges
- To take a snapshot, e.g. from **psql**:

```
SELECT pgstatviz.snapshot();
```

Announcing pg_statviz

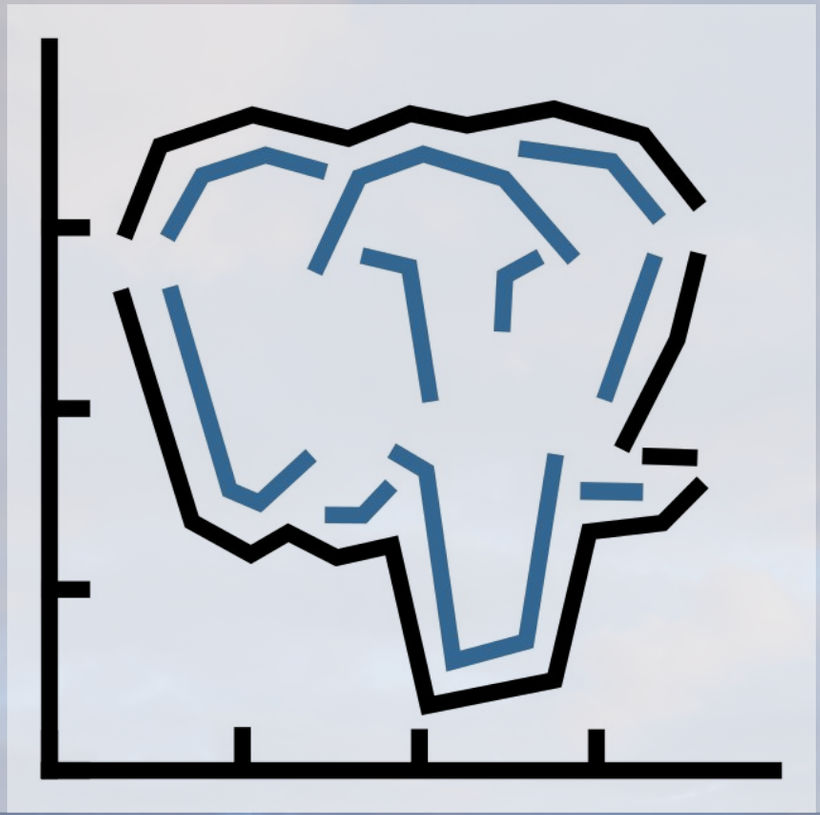
```
vyruss@rancor:~$ pg_statviz --help
usage: pg_statviz [--help] [--version] [-d DBNAME] [-h HOSTNAME] [-p PORT] [-U USERNAME] [-W]
                  [-D FROM TO] [-O OUTPUTDIR]
                  {analyze,buf,cache,checkp,conn,tuple,wait,wal} ...

run all analysis modules

positional arguments:
  {analyze,buf,cache,checkp,conn,tuple,wait,wal}
    analyze             run all analysis modules
    buf                 run buffers written analysis module
    cache               run cache hit ratio analysis module
    checkp              run checkpoint analysis module
    conn                run connection count analysis module
    tuple               run tuple count analysis module
    wait                run wait events analysis module
    wal                 run WAL generation analysis module

options:
  --help
  --version             show program's version number and exit
  -d DBNAME, --dbname DBNAME
                        database name to analyze (default: 'vyruss')
  -h HOSTNAME, --host HOSTNAME
                        database server host or socket directory (default: '/var/run/postgresql')
  -p PORT, --port PORT  database server port (default: '5432')
  -U USERNAME, --username USERNAME
                        database user name (default: 'vyruss')
  -W, --password        force password prompt (should happen automatically) (default: False)
  -D FROM TO, --daterange FROM TO
                        date range to be analyzed in ISO 8601 format e.g. 2023-01-01T00:00
                        2023-01-01T23:59 (default: [])
  -O OUTPUTDIR, --outputdir OUTPUTDIR
                        output directory (default: -)
vyruss@rancor:~$
```

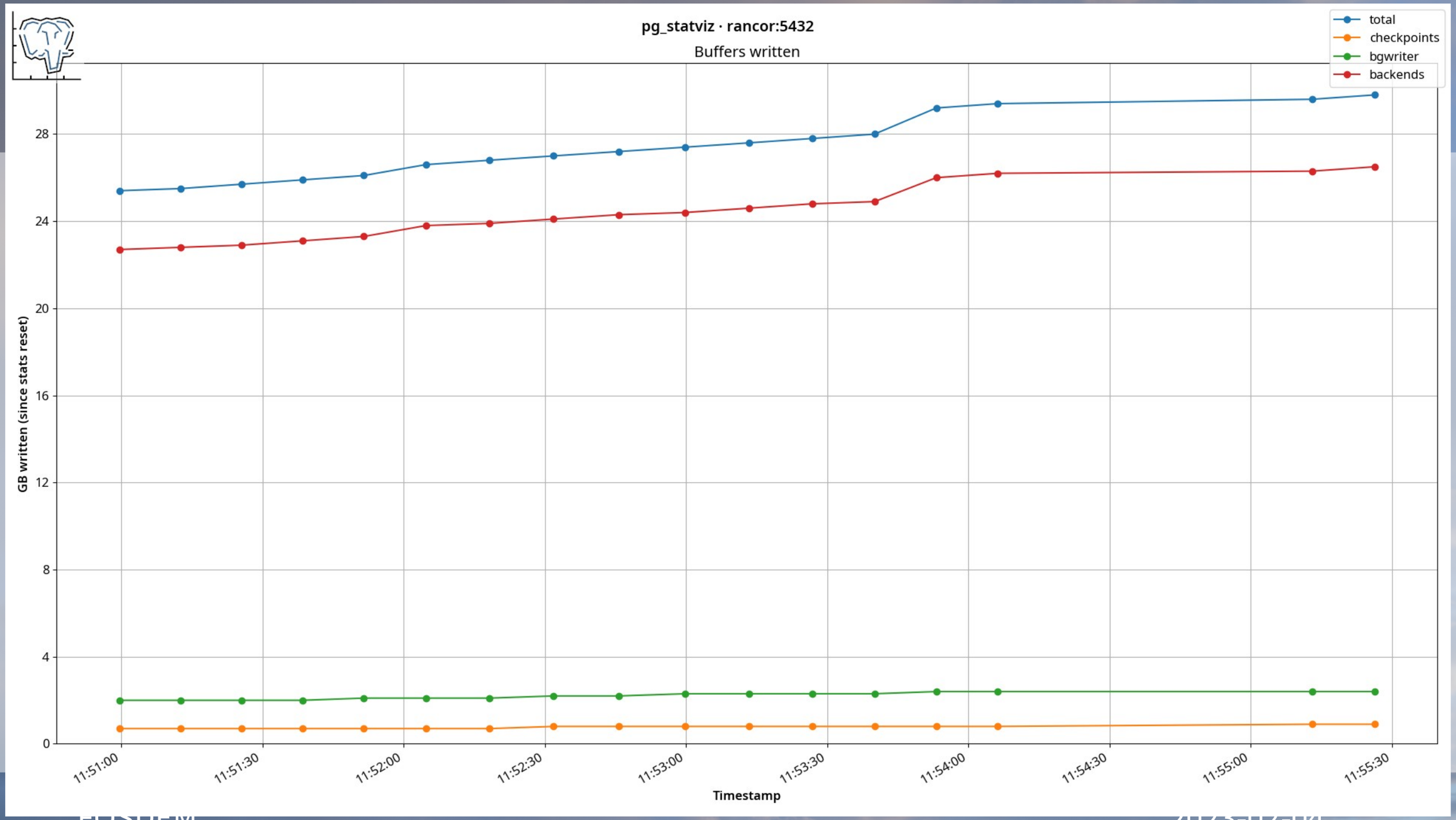
Announcing pg_statviz

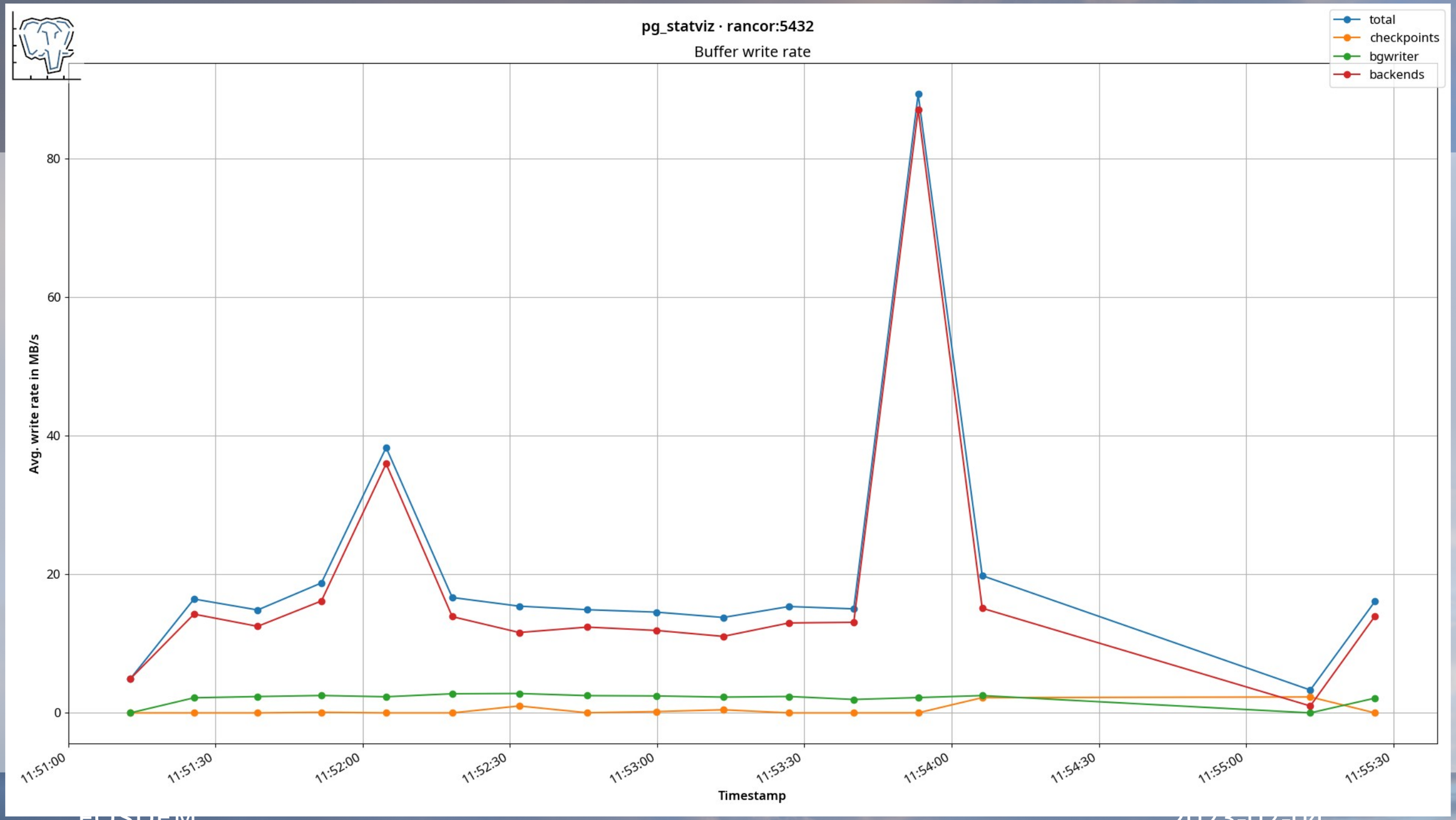
```
vyruss@rancor:~$ pg_statviz -d faf -U postgres -D 2023-01-17T23:00 2024-01-01
INFO:pg_statviz.modules.buf:Running buffers written analysis
INFO:pg_statviz.modules.buf:Saving pg_statviz_rancor_5432_buf.png
INFO:pg_statviz.modules.buf:Saving pg_statviz_rancor_5432_buf_rate.png
INFO:pg_statviz.modules.checkp:Running checkpoint analysis
INFO:pg_statviz.modules.checkp:Saving pg_statviz_rancor_5432_checkp.png
INFO:pg_statviz.modules.checkp:Saving pg_statviz_rancor_5432_checkp_rate.png
INFO:pg_statviz.modules.cache:Running cache hit ratio analysis
INFO:pg_statviz.modules.cache:Saving pg_statviz_rancor_5432_cache.png
INFO:pg_statviz.modules.conn:Running connection count analysis
INFO:pg_statviz.modules.conn:Saving pg_statviz_rancor_5432_conn_status.png
INFO:pg_statviz.modules.conn:Saving pg_statviz_rancor_5432_conn_user.png
INFO:pg_statviz.modules.tuple:Running tuple count analysis
INFO:pg_statviz.modules.tuple:Saving pg_statviz_rancor_5432_tuple.png
INFO:pg_statviz.modules.wait:Running wait events analysis
INFO:pg_statviz.modules.wait:Saving pg_statviz_rancor_5432_wait.png
INFO:pg_statviz.modules.wal:Running WAL generation analysis
INFO:pg_statviz.modules.wal:Saving pg_statviz_rancor_5432_wal.png
INFO:pg_statviz.modules.wal:Saving pg_statviz_rancor_5432_wal_rate.png
vyruss@rancor:~$
```

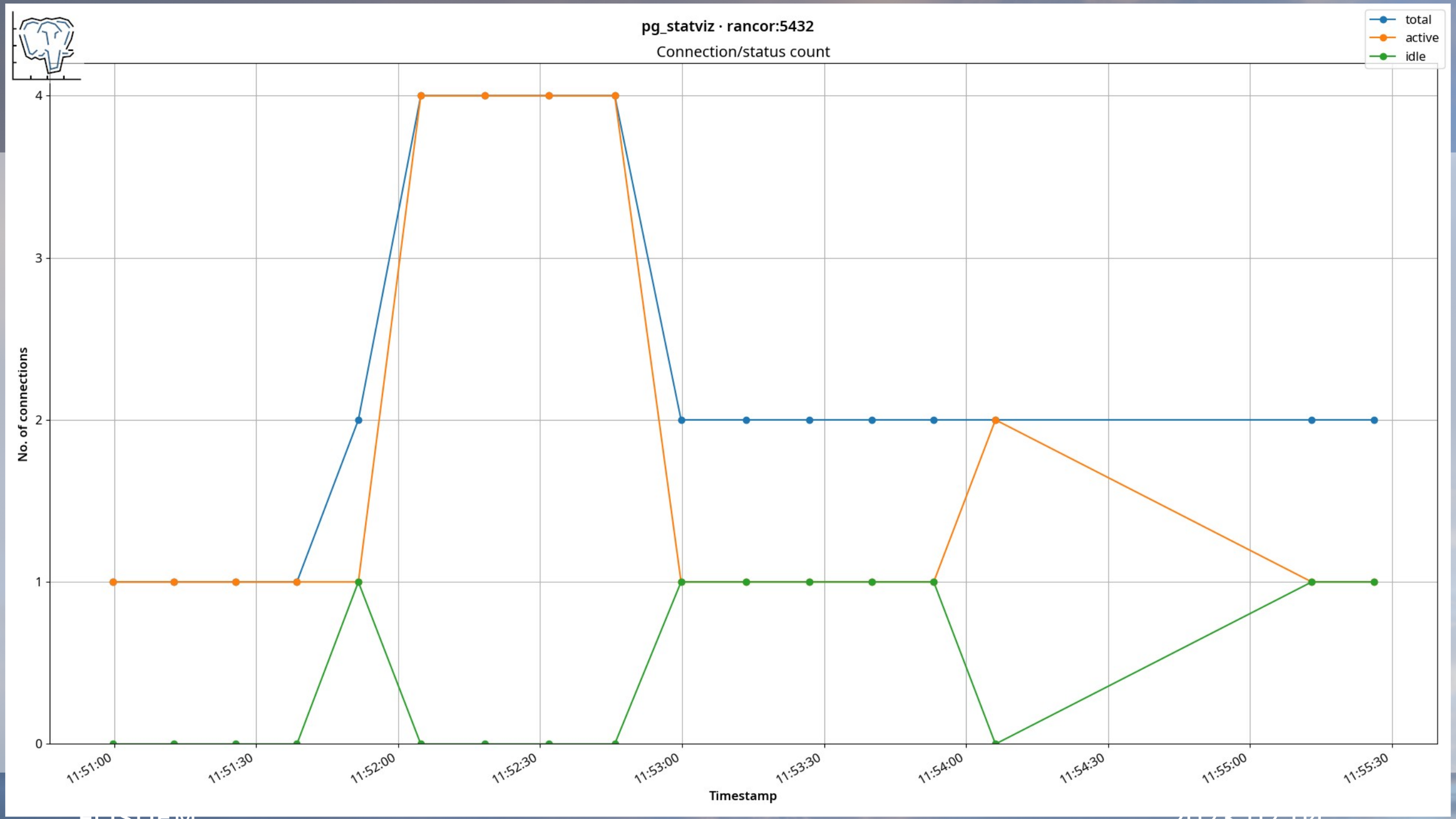
Announcing pg_statviz

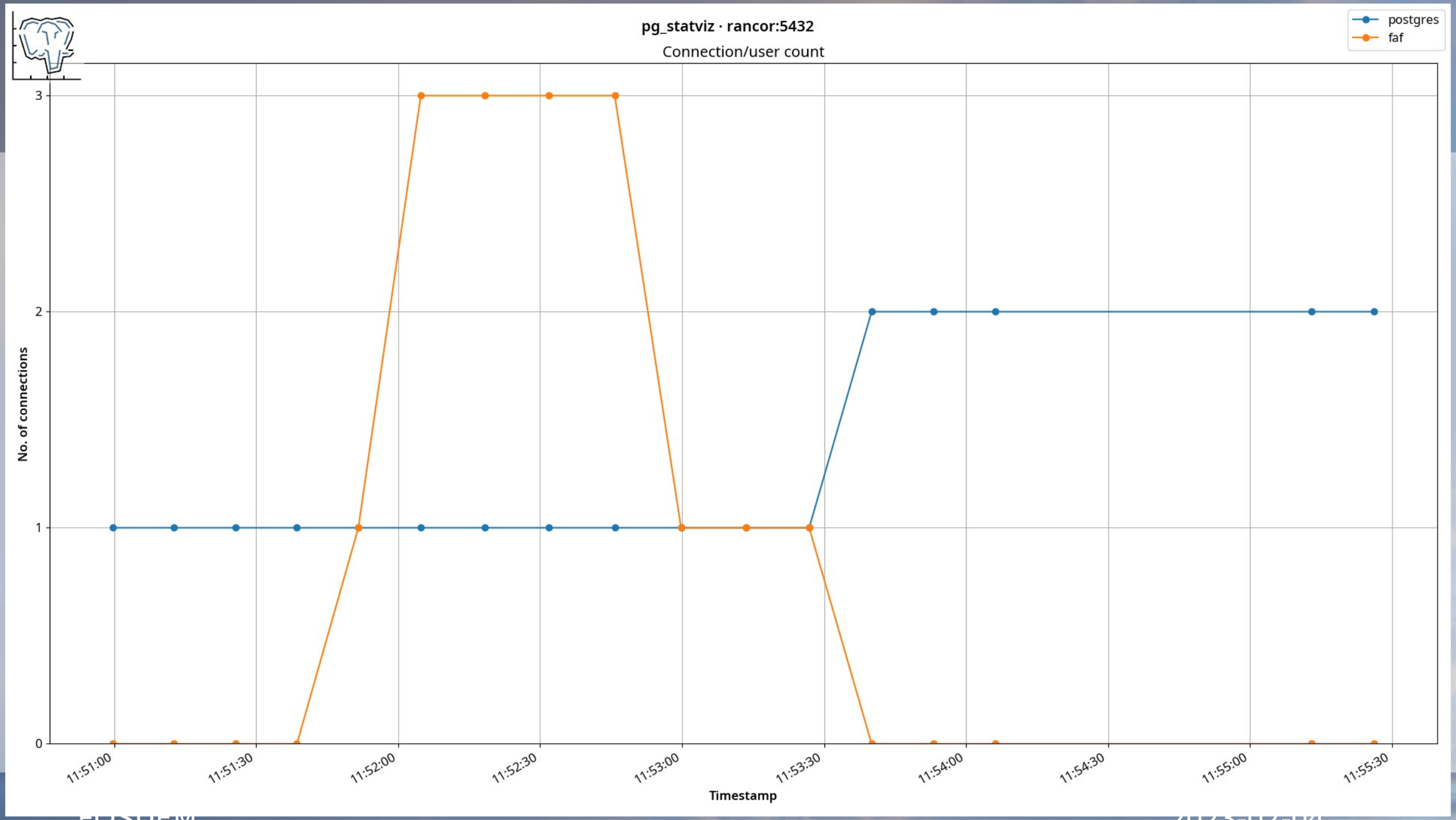
- And yes, it has a logo :)

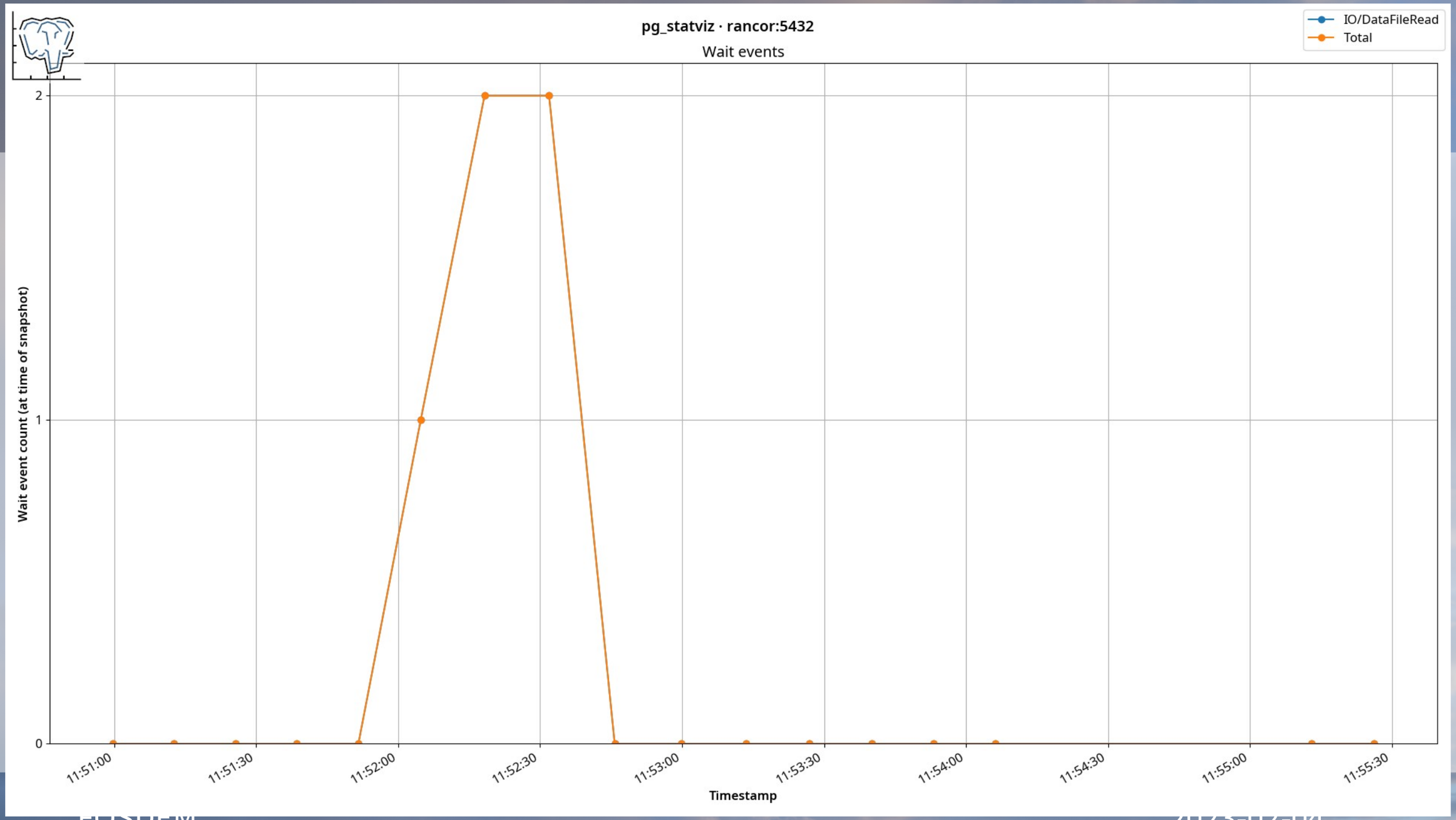



pg_statviz · rancor:5432
Buffers written
total
checkpoints
bgwriter
backends
GB written (since stats reset)
0
4
8
12
16
20
24
28
11:51:00
11:51:30
11:52:00
11:52:30
11:53:00
11:53:30
11:54:00
11:54:30
11:55:00
11:55:30
Timestamp


pg_statviz · rancor:5432
Buffer write rate
total
checkpoints
bgwriter
backends
Avg. write rate in MB/s
0
20
40
60
80
11:51:00
11:51:30
11:52:00
11:52:30
11:53:00
11:53:30
11:54:00
11:54:30
11:55:00
11:55:30
Timestamp


pg_statviz · rancor:5432
Connection/status count
total
active
idle
No. of connections
0
1
2
3
4
11:51:00
11:51:30
11:52:00
11:52:30
11:53:00
11:53:30
11:54:00
11:54:30
11:55:00
11:55:30
Timestamp


pg_statviz · rancor:5432
Connection/user count
postgres
faf
No. of connections
0
1
2
3
11:51:00
11:51:30
11:52:00
11:52:30
11:53:00
11:53:30
11:54:00
11:54:30
11:55:00
11:55:30
Timestamp


pg_statviz · rancor:5432
Wait events
IO/DataFileRead
Total
Wait event count (at time of snapshot)
2
1
0
11:51:00
11:51:30
11:52:00
11:52:30
11:53:00
11:53:30
11:54:00
11:54:30
11:55:00
11:55:30
Timestamp

Announcing pg_statviz

- Thank you!
- Project:

  https://github.com/vyruss/pg_statviz

- You can find me on Mastodon:

  @vyruss@fosstodon.org