

2016-05-19



## Elephants and Raft

Constantin S. Pan  
kvapen@gmail.com

## Reasons

- ▶ We needed a replicated key-value storage for
  - ▶ distributed deadlock detection;
  - ▶ node failure detection.
- ▶ We had Raft implemented in C (for the distributed transaction daemon).

## Extension

Raft-based key-value storage as a Postgres extension.  
[https://github.com/postgrespro/postgres\\_cluster](https://github.com/postgrespro/postgres_cluster),  
look in contrib/raftable.

1. Add these to the config:

```
shared_preload_libraries = raftable
raftable.id = 1
raftable.peers = 1:ip:port,2:ip:port,...
```

2. create extension raftable;

## Available in SQL

```
-- set
select raftable('my key', 'my value');

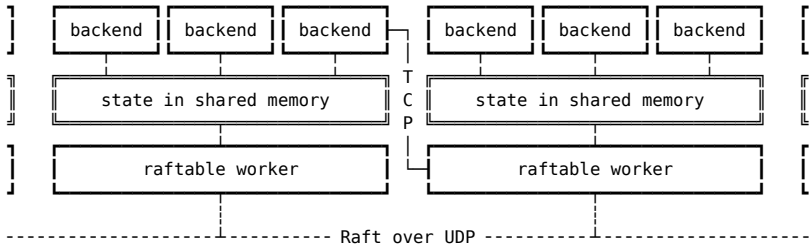
-- get
select raftable('my key');

-- list
select * raftable();
```

## Available in C

```
/* set */  
raftable_set("my key", "val", strlen("val"));  
  
/* get */  
char *myval = raftable_get("my key", &vallen);  
  
/* iterate */  
raftable_every(myfunc, myarg);  
/* calls myfunc(key, value, size, myarg)  
   for every pair */
```

# Structure



## TODO

- ▶ Tests
- ▶ Node management