Quick Start Guide PgBouncer

By CJ Estel

What is PgBouncer

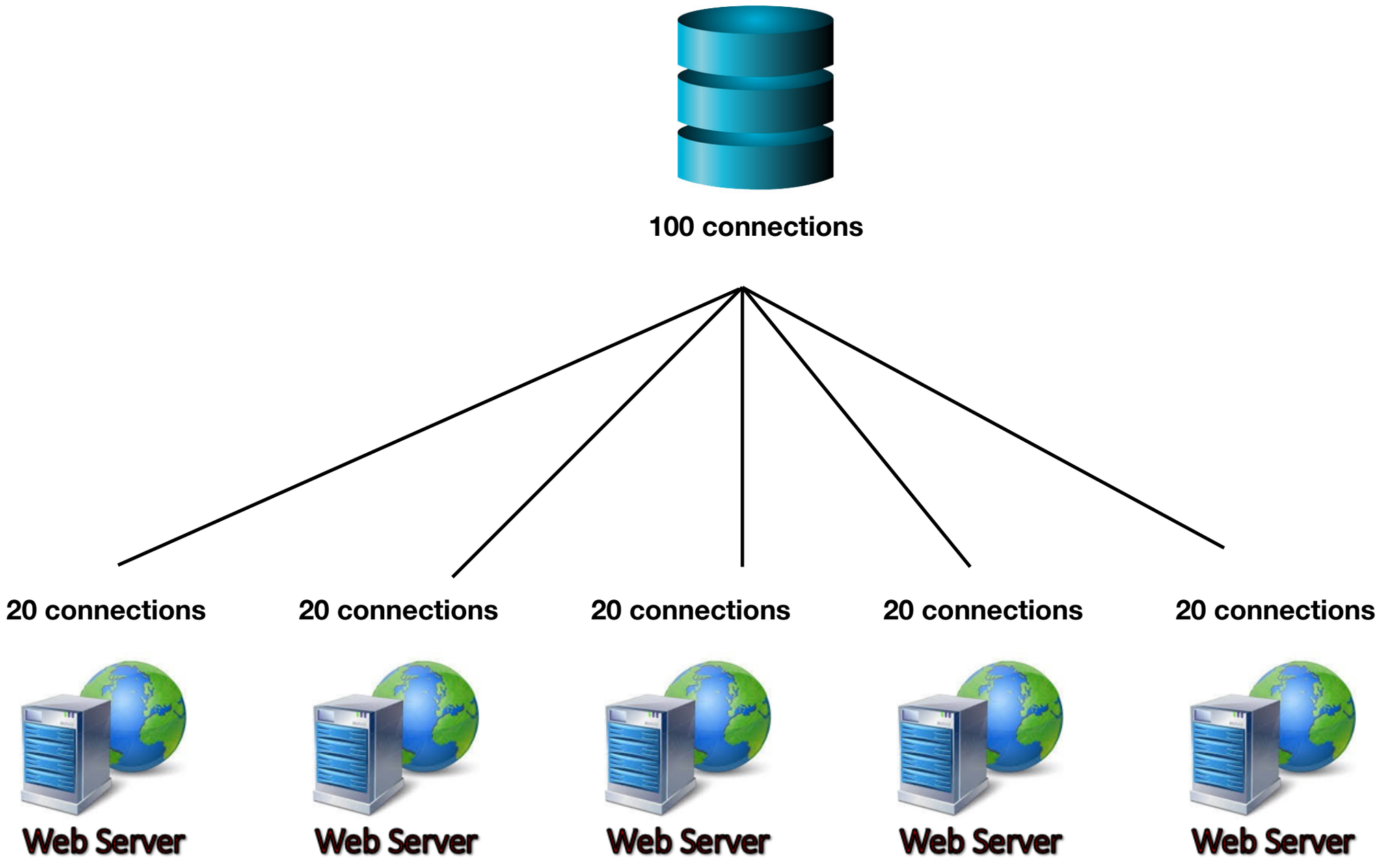
- PgBouncer is a connections pooling service for Postgres. It has all kinds of internal limits and limited resources.

- Some Useful Links
  - https://pgbouncer.github.io/config.html
  - https://hunleyd.github.io/posts/pgBouncer-and-auth-pass-thru/

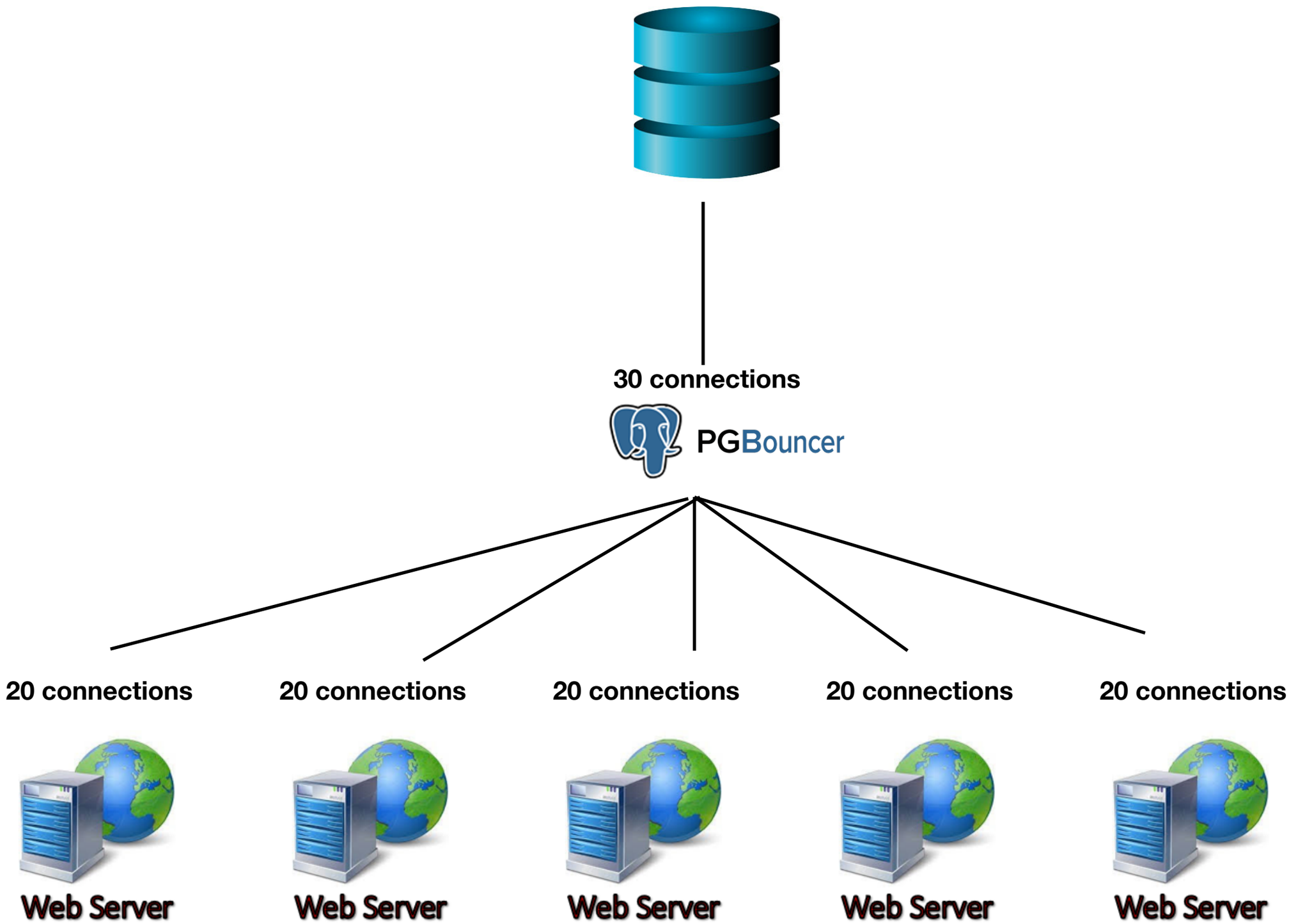
Why Do I Need a Connection Pooler?

- Opening and Closing connections can be expensive
- Moar connections means moar resource overhead
- PgBouncer can assist in relatively seamless failover situations

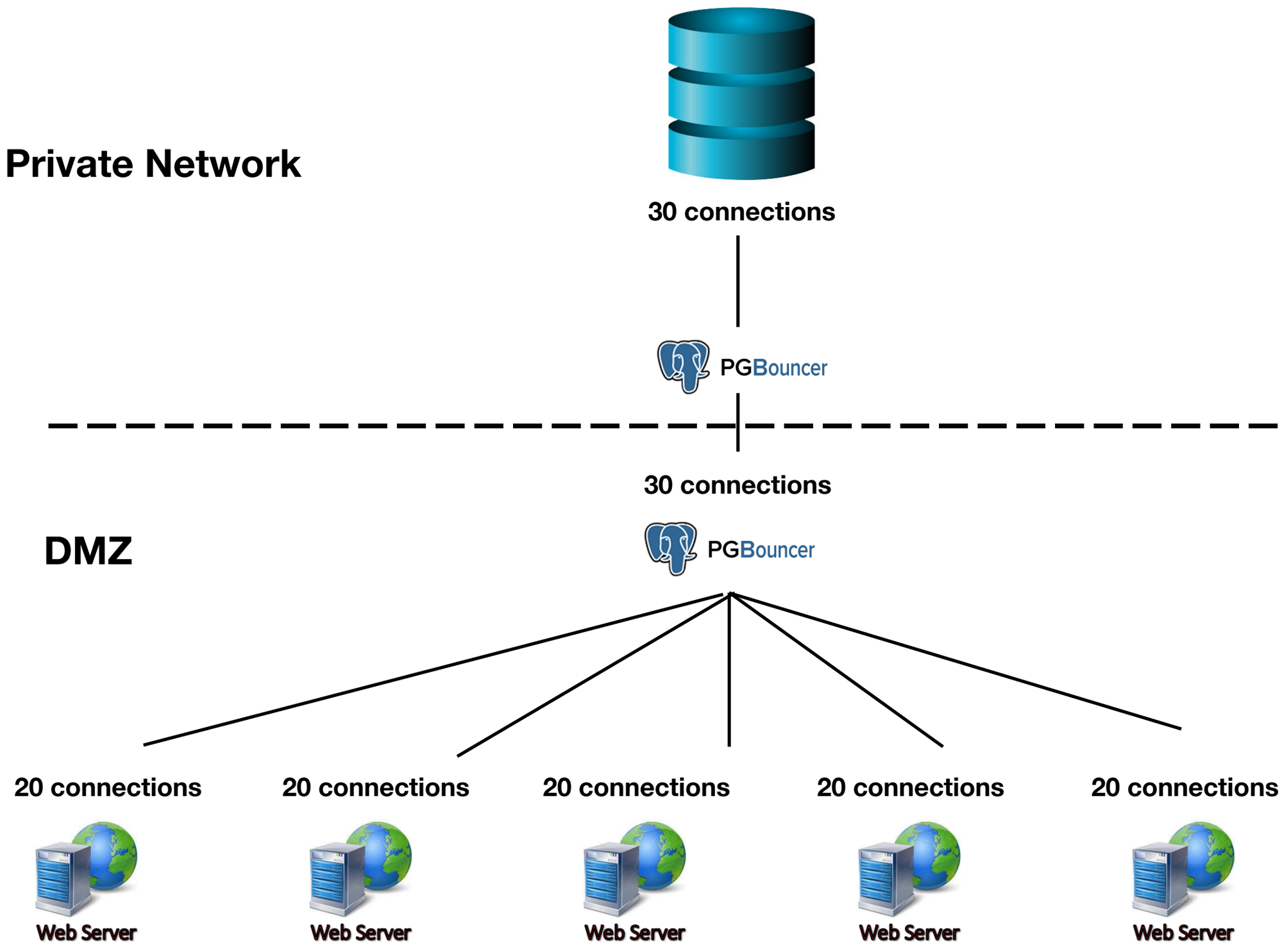
Example Without Pooling



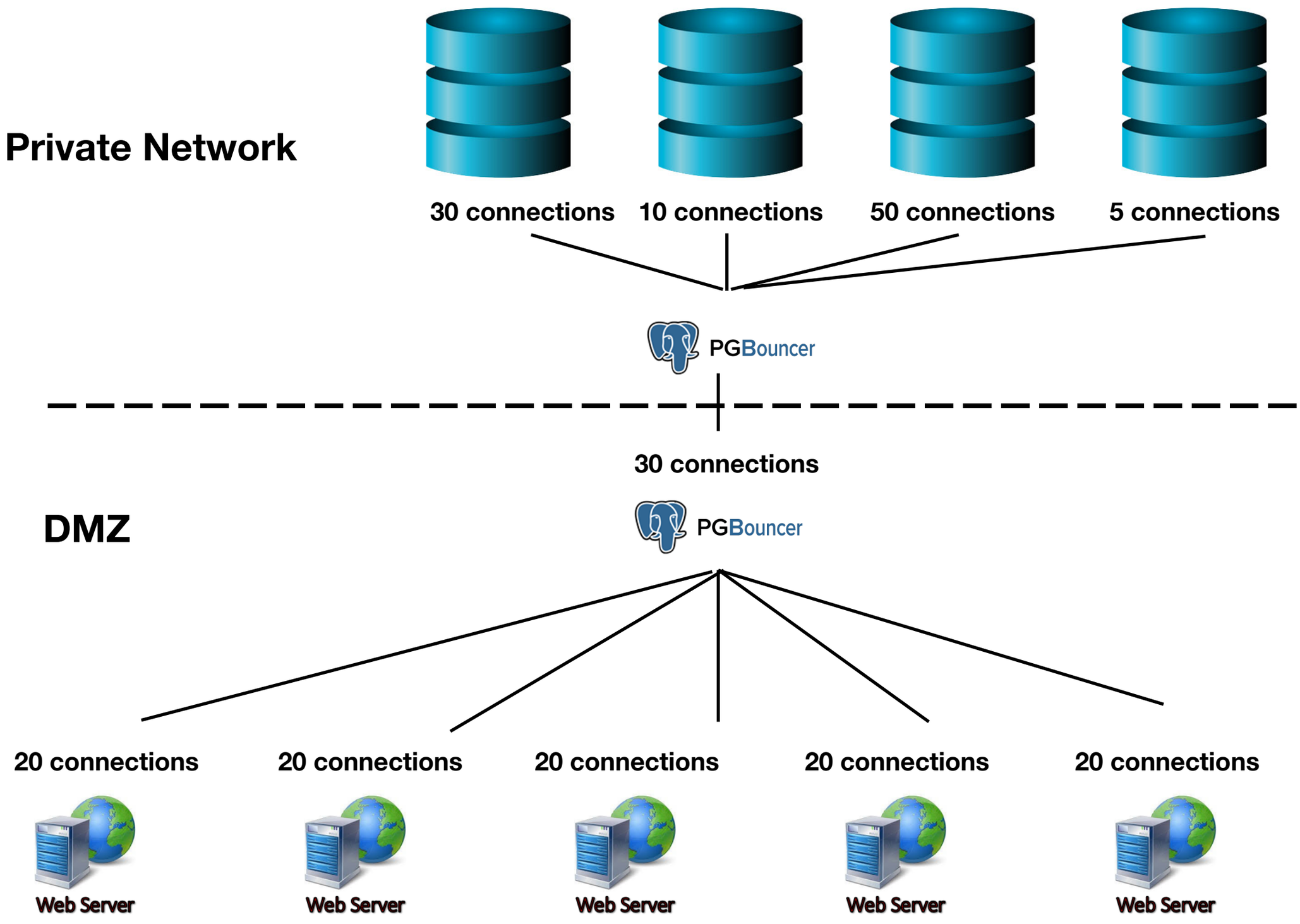
Example With Pooling



Example With Two Layers



Example As Data Router



Types of Pooling (pool_mode)

- Session
- Transaction
- Statement

Feature	Session pooling [1]	Transaction pooling
Startup parameters [2]	Yes	Yes
SET/RESET	Yes	No
LISTEN/NOTIFY	Yes	No
WITHOUT HOLD CURSOR	Yes	Yes
WITH HOLD CURSOR	Yes	No
Protocol-level prepared plans	Yes	No [3]
PREPARE / DEALLOCATE	Yes	No
ON COMMIT DROP temp tables	Yes	Yes
PRESERVE/DELETE ROWS temp tables	Yes	No
Cached plan reset	Yes	Yes [1]
LOAD statement	Yes	No
UDFs with session state	Yes	No

Failover

- Ability to “pause” connections
- Ability to reload config to point to a new server
- Easy to hook into from other failover tools (repmgr, ansible, consul, etc)

Configuration

- /etc/pgbouncer
  - pgbouncer.ini
    - Lists databases and options
- userlist.txt
  - Lists users

pgbouncer.ini

```
[databases]

sessions_cmmphp = host=[hostname] dbname=[database_name] user=[username]

[pgbouncer]

logfile = /var/log/pgbouncer/pgbouncer.log
pidfile = /var/run/pgbouncer/pgbouncer.pid
unix_socket_dir = /tmp
auth_file = /etc/pgbouncer/userlist.txt
listen_addr = *
listen_port = 5432
admin_users = admin
stats_users = admin
auth_type = md5
pool_mode = session
server_reset_query = DISCARD ALL
server_check_query = select 1
server_check_delay = 30
max_client_conn = 1000
default_pool_size = 50
server_tls_sslmode = prefer
min_pool_size = 5
reserve_pool_size = 10
log_connections = 0
log_disconnections = 0
```

Authentication Methods

pam
PAM is used to authenticate users, auth_file is ignored. This method is not compatible with databases using auth_user option. Service name reported to PAM is “pgbouncer”. Also, pam is still not supported in HBA configuration file.

hba
Actual auth type is loaded from auth_hba_file. This allows different authentication methods different access paths. Example: connection over Unix socket use `peer` auth method, connection over TCP must use TLS. Supported from version 1.7 onwards.

cert
Client must connect over TLS connection with valid client cert. Username is then taken from CommonName field from certificate.

md5
Use MD5-based password check. auth_file may contain both MD5-encrypted or plain-text passwords. This is the default authentication method.

plain
Clear-text password is sent over wire. Deprecated.

trust
No authentication is done. Username must still exist in auth_file.

any
Like the `trust` method, but the username given is ignored. Requires that all databases are configured to log in as specific user. Additionally, the console database allows any user to log in as admin.

userlist.txt

```
"admin" "md5xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx"
"php_rw" "md5xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx"
```

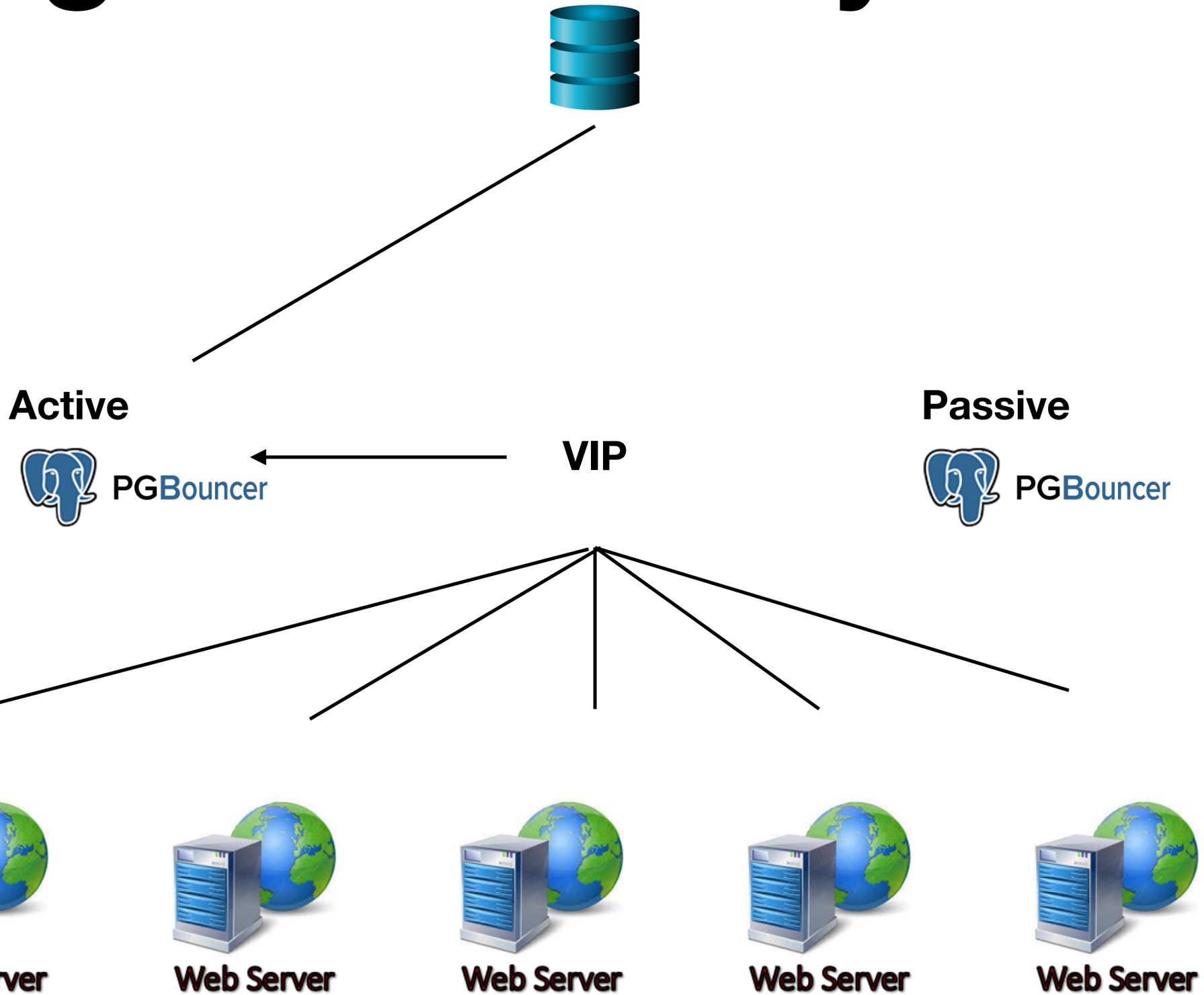
Commands

```
psql -U admin -h localhost -d pgbouncer

Help output:

pgbouncer=# show help;
NOTICE:  Console usage
DETAIL:
        SHOW HELP|CONFIG|DATABASES|POOLS|CLIENTS|SERVERS|VERSION
        SHOW STATS|FDS|SOCKETS|ACTIVE_SOCKETS|LISTS|MEM
        SHOW DNS_HOSTS|DNS_ZONES
        SET key = arg
        RELOAD
        PAUSE [<db>]
        RESUME [<db>]
        DISABLE <db>
        ENABLE <db>
        KILL <db>
        SUSPEND
        SHUTDOWN
SHOW
```

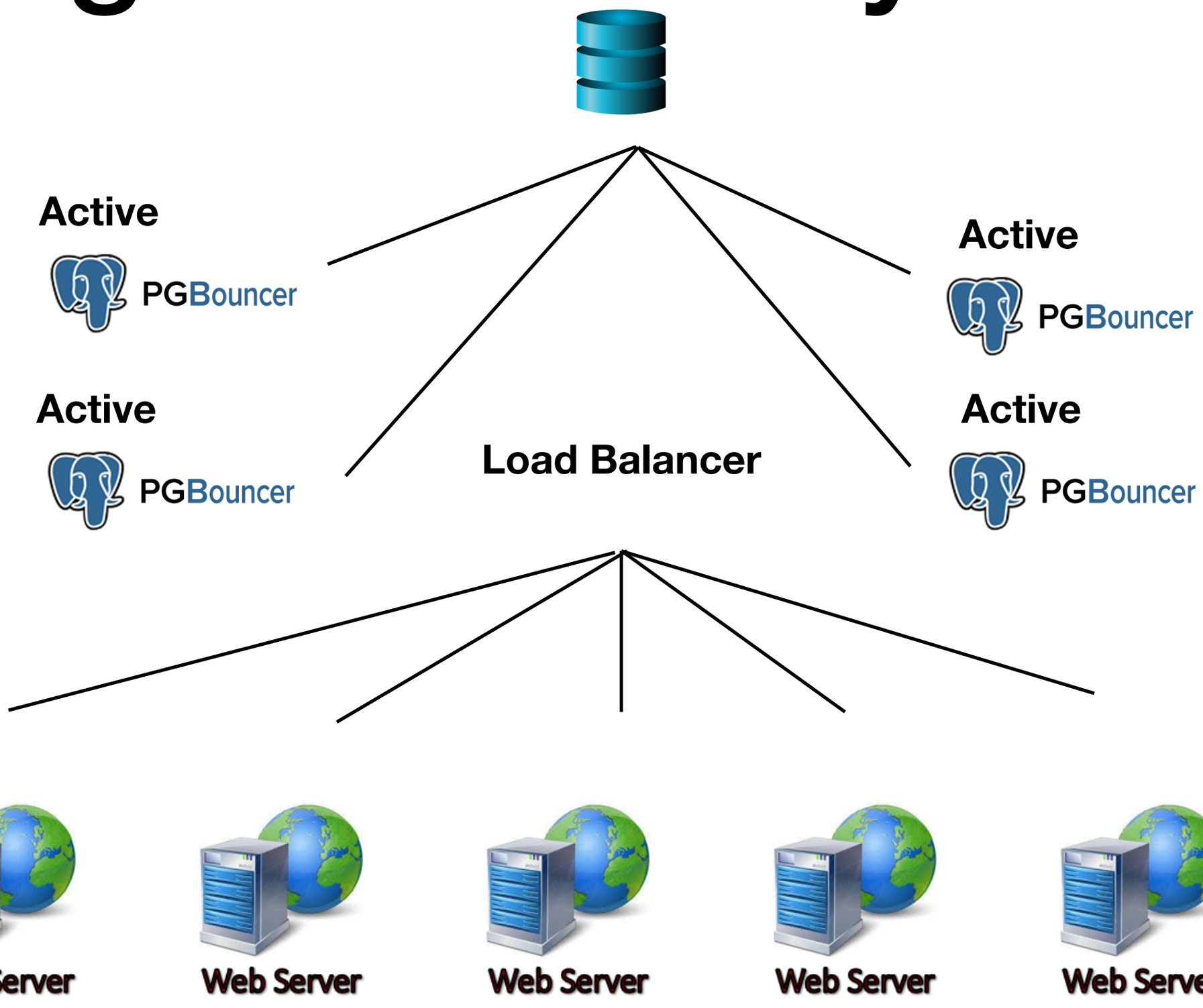
watch "psql -h localhost -U admin -d pgbouncer -c 'show pools';"

```
Every 2.0s: psql -h localhost -U admin -d pgbouncer -c 'show pools';                                                        Mon Jun 24 21:25:58 2019

    database     |     user     | cl_active | cl_waiting | sv_active | sv_idle | sv_used | sv_tested | sv_login | maxwait | maxwait_us | pool_mode
-----------------+--------------+-----------+------------+-----------+---------+---------+-----------+----------+---------+------------+-----------
 pgbouncer       | pgbouncer    |         1 |          0 |         0 |       0 |       0 |         0 |        0 |       0 |          0 | statement
 [redacted]php | [redacted]wr162 |         0 |          0 |         0 |       4 |       1 |         0 |        0 |       0 |          0 | session
(2 rows)
```

High Availability Setup



High Availability Setup



How Do I Monitor It?

- https://bucardo.org/check_postgres/
  - pgb_pool_cl_{active,**waiting**}
  - pgb_pool_sv_{**active**,idle,used,tested,login}
  - **pgb_pool_maxwait**
  - pgbouncer_backends
  - **pgbouncer_checksum**
- Vividcortex