Patroni

The real life experience and migration path

Jan Tomsa

Já

- Jan Tomsa
- DevOps v Showmaxu
- jan.tomsa@showmax.com

Co je Showmax?

- VOD (Video On Demand) služba
  - koncept mikroslužeb (máme rádi kontejnery)
  - EU (datacentrum) + Afrika (zákazníci)
  - Vývoj v Praze a Berouně
  - Máme rádi PostgreSQL :)

https://kinkyrhino.co.za/product/showmax-and-chill

S čím jsme začínali

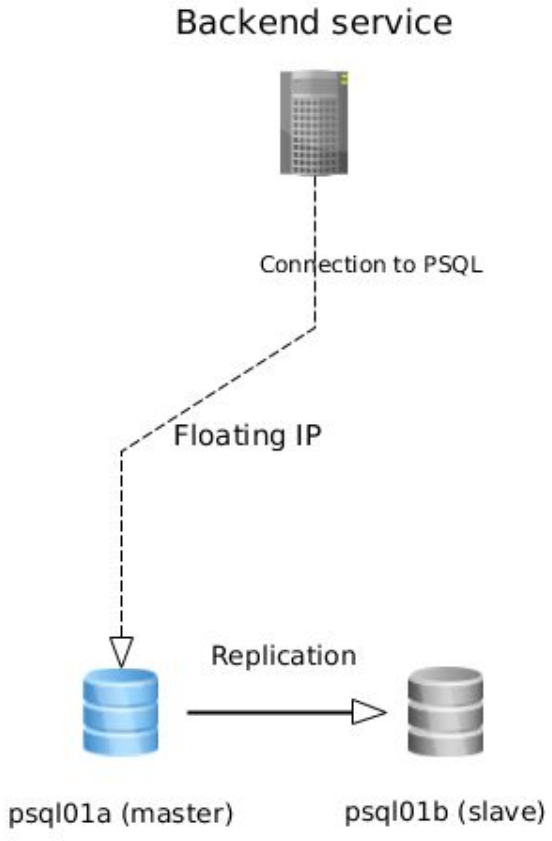
- 7 produkčních (+1 stage) aplikačních PSQL clusterů
  - ~600 GB
- 1 data warehouse
  - 2 TB
- 1 analytický
  - 1 TB
- 1 pro zbytek infrastruktury

Původní topologie

- Master -> slave replikace (repmgr)
- “Floating IP”
  - Mířící na aktuálního mastera
  - Není to DNS failover,
    ale extra IP routovaná na libovolný stroj

\+ Fungovalo to :D

\- Manuální failover
  - Master <-> Slave
  - IP (pomocí API callu - management datacentra)



Dlouhodobě neudržitelné

- Pomalý failover s dlouhým výpadkem
- Noční zásahy on-callisty
- Jen jedna replika

http://2.bp.blogspot.com/-nHW-TIx6TKk/U9Kw8Fi_tqI/AAAAAAAALFg/gfJGhbb1bQE/s1600/unbearable.jpg

Potřebujeme automatizované řešení

- “Bezzásahové” a automatizované řešení
- Rychlý a jednoduchý (i manuální) failover

https://dynamicmedia.zuza.com/zz/m/original_/8/a/8a64a122-8ec3-43e3-9f51-93054360bcff/IYN4_SS__Super_Portrait.jpg

Kandidáti

- 2 kandidáti
  - Podobné vlastnosti a architektura
- Patroni (https://github.com/zalando/patroni) - python, HAProxy
- Stolon (https://github.com/sorintlab/stolon) - go, vlastní proxy

Patroni

- Vyvíjen Zalando
- Šablona pro HA PostgreSQL
  - Velké možnosti přizpůsobení
- Python “wrapper” nad PostgreSQL
- Stará se o celý životní cyklus postgresu

Odkud získat Patroni?

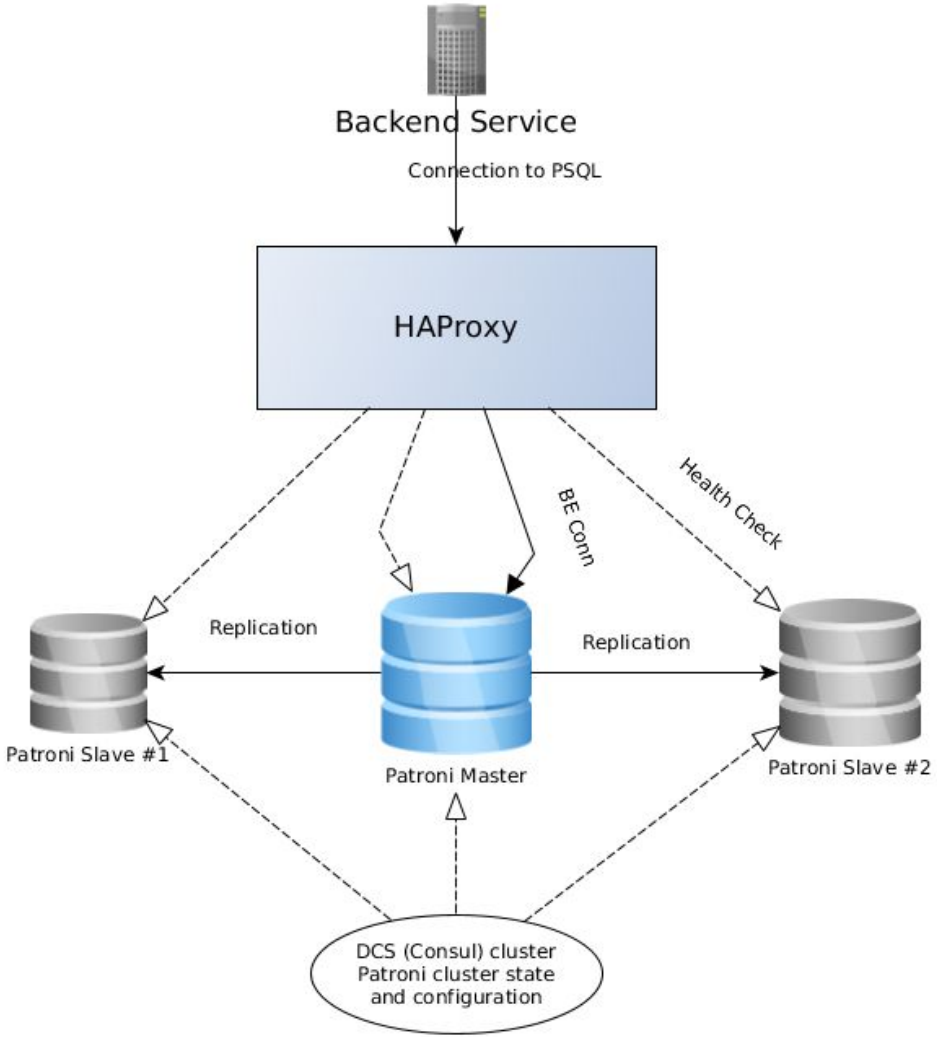
- Zdrojový kód
  - GitHub - https://github.com/zalando/patroni
- Dokumentace
  - https://patroni.readthedocs.io/en/latest/
- Balíčky
  - PIP - https://pypi.org/project/patroni/
  - DEB - repozitář PostgreSQL https://www.postgresql.org/download/

První krůčky - testovací cluster v Docker

- Naklonovat repo - https://github.com/zalando/patroni
  - Obsahuje Dockerfile a Docker compose file
- `docker-compose up`
- Spustí
  - ETCD
  - 3x Patroni s PostgreSQL
  - HAProxy

https://img.csfd.cz/files/images/film/photos/000/081/81627_08378a.jpg

Topologie


Backend Service
Connection to PSQL
HAProxy
BE Conn
Health Check
Replication
Replication
Patroni Slave #1
Patroni Master
Patroni Slave #2
DCS (Consul) cluster
Patroni cluster state
and configuration

Nástroje - REST API

- Zjišťování stavu
  - Využíváno např. HAProxy pro routing
    - API vrátí 200 nebo 503

```
curl -v -XOPTIONS localhost:8008/master
curl -v -XOPTIONS localhost:8008/replica
```

- Konfigurace
  - čtení a nastavení tzv dynamické konfigurace clusteru

Nástroje - PATRONICTL

- Commandlinový nástroj pro management clusteru
- failover/switchover
- Reinicializace repliky
- Stav clusteru (pretty, json, yaml, …)

```
 root@patroni01a:~# patronictl -c /etc/patroni/patroni.yml list
+---------------+------------+--------------------------+--------+---------+-----------+-----------------+
|    Cluster    |   Member   |           Host           |  Role  |  State  | Lag in MB | Pending restart |
+---------------+------------+--------------------------+--------+---------+-----------+-----------------+
| patroni01-p11 | patroni01a | patroni01a               | Leader | running |       0.0 |        *        |
| patroni01-p11 | patroni01b | patroni01b               |        | running |       0.0 |        *        |
| patroni01-p11 | patroni01c | patroni01c               |        | running |       0.0 |        *        |
+---------------+------------+--------------------------+--------+---------+-----------+-----------------+
```

Základní možnosti konfigurace

Tři typy konfigurace

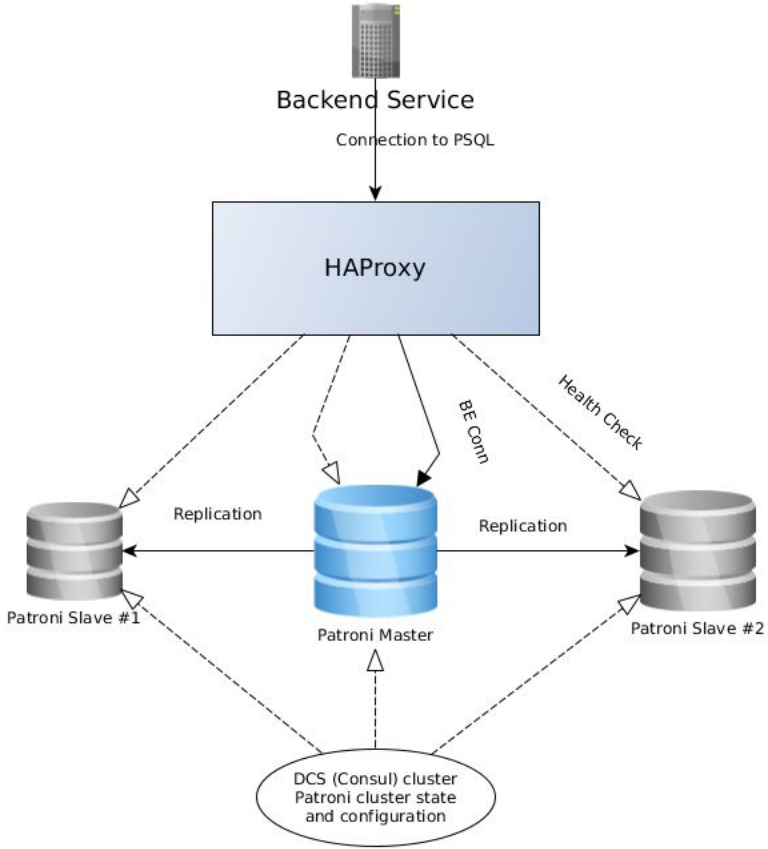
- Dynamická konfigurace
  - uložená v DCS (consul, ETCD, …)
- Lokální konfigurace
- Konfigurace pomocí proměnných prostředí

Základní možnosti konfigurace

- Patroni.yml
  - Hlavní konfigurační soubor
  - Nastavení globální i lokální konfigurace
  - Nastavení parametrů clusteru (timeouty, …)
  - Bootstrap (parametry initdb, …)
  - Nastaveni postgresu
    - Hba, config postgresu, superuser, ...

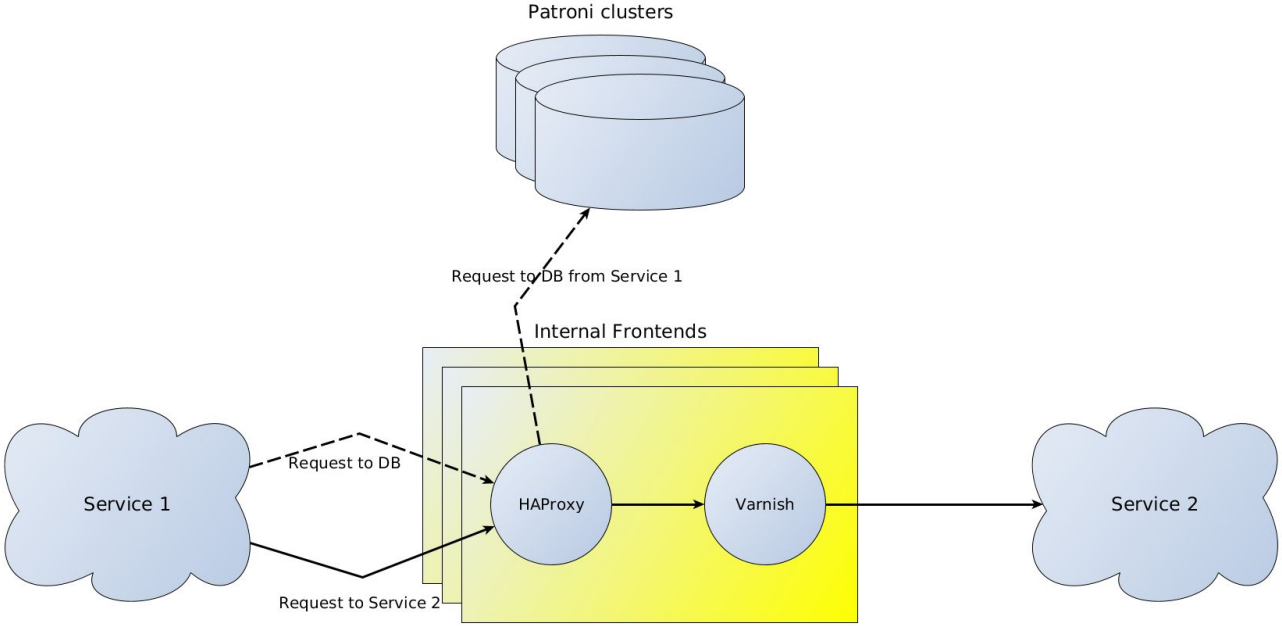
Patroni @ Showmax

- 9 clusterů,
  Z toho 6 produkčních
- Cluster: 3 dedikovane servery
- DCS: consul (5 node cluster)
- Proxy: HAProxy
- Zálohy: PgDump + Barman
- Monitoring: Icinga + Prometheus



Patroni @ Showmax - HAProxy

- Mikroservisy spolu komunikují prostřednictvím HAProxy a Varnish



Patroni @ Showmax - HAProxy

```
frontend patroni01_master
    mode tcp
    timeout client          40m
    bind <bind_ip>:5000
    default_backend bk_patroni01_master

backend bk_patroni01_master
    mode tcp
    timeout server          40m
    timeout connect         500
    option httpchk OPTIONS /master
    server <hostname1> <ip1>:5432 maxconn 500 check port 8008 inter 1000
    server <hostname2> <ip2>:5432 maxconn 500 check port 8008 inter 1000
    server <hostname3> <ip3>:5432 maxconn 500 check port 8008 inter 1000
```

Patroni @ Showmax - Backup - pgdump

- Pravidelný dump každou noc
- Pgdump se připojuje
  na master pomocí HAProxy

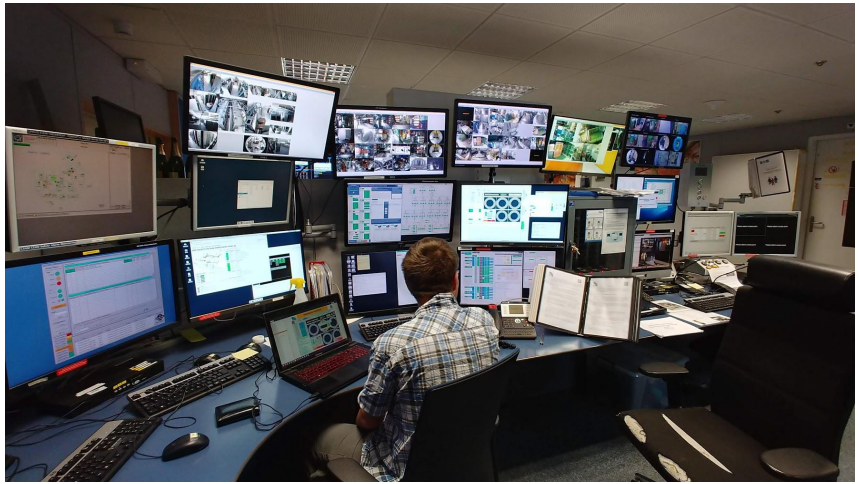
https://commons.wikimedia.org/wiki/File:Dump_Truck_Dumping_Toxic_Medical_Waste.png

Patroni @ Showmax - Backup - Barman

- Barman https://www.pgbarman.org/
  - “administration tool for disaster recovery of PostgreSQL”
  - Streamování WALů (archive_command)
  - pravidelný base_backup
  - možnost PIT recovery
- Adaptace na Patroni:
  1. Nakonfigurovat Barman pro všechny nody v clusteru
  2. V go psaný wrapper spouští barman backup proti aktuálnímu masteru

Patroni @ Showmax - monitoring

- Icinga
  - Aktivní checky - test připojení do všech db
  - Monitoring stavu backendů HAProxy
- Prometheus
  - Sběr statistik o stavu clusteru a alerting
    - Přepínání masterů
    - Opožděnost replik za masterem
    - Aktuální počet konexí na server



Trable s Patroni @ Showmax
aneb jak Vás můžou Patroni postřelit

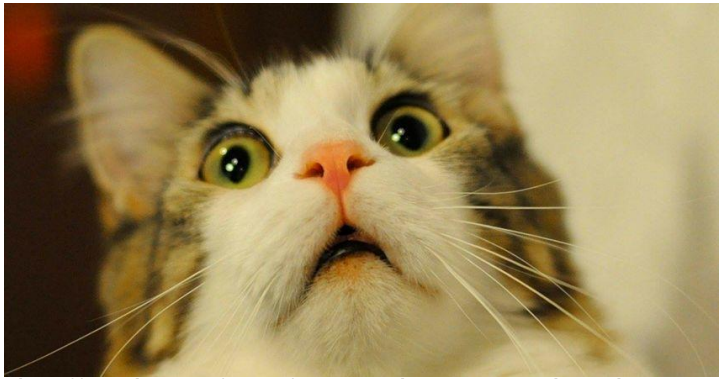
https://geekandsundry.com/the-wait-is-over-discover-your-patronus-with-pottermores-new-quiz.jpg

https://pixabay.com/en/patrons-bullets-sleeves-ammo-1493479.png

Kaskádové neštěstí

- Po migraci na jiný HW
- Noční backup - master přestal odpovídat
  - Patroni zafungovaly a jedna z replik se stala masterem
- O pár dní později stejná situace
  - Ale zemřely stejným způsobem 2 ze 3 nod
- Vyšetřování
  - Kernel panic
  - Upgrade kernelu
  - Snaha chytit trace pomoci netconsole

Kaskádové neštěstí

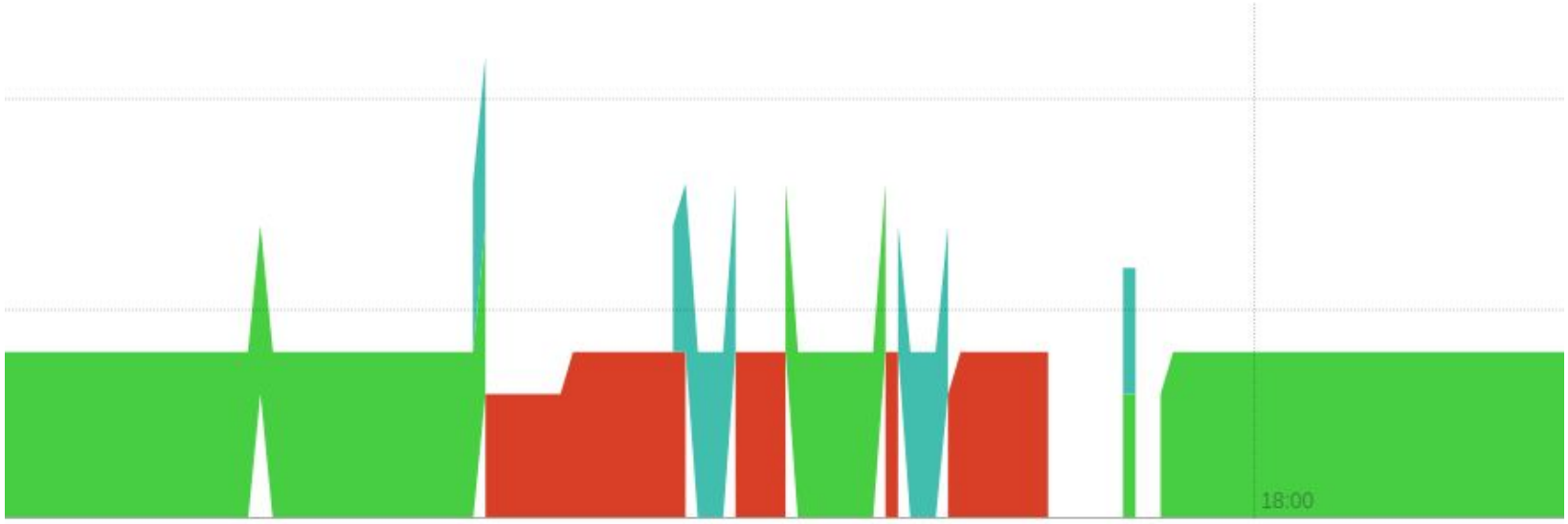


https://www.iheart.com/content/2017-11-10-does-your-cat-see-ghosts-why-your-pet-stares-into-space/.jpeg

- Podvečerní špička
  - Servery začaly porůznu mizet z patronictl
  - Chvílemi byl v clusteru jen jeden server
  - Patroni si během toho stěžovaly že nemohou komunikovat s DCS (consul)
- Very scary situation
  - Měli jsme plné ruce práce udržet alespoň jeden node živý
  - Naštěstí se skoro vždy podařilo reinicializovat nějaký slave, než aktuální master vypadl
  - Uživatelské requesty failovaly - servisy se stále musely přepojovat na jinou DB

Kaskádové neštěstí

- Záznam přepínání masteru



Co se vlastně stalo?

- Nekvalitní síťová karta (Intel i219LM, použitý driver e1000)
  - Zátěž se zapnutým offloadingem - nestabilní
  - Časté restarty - “Reset adapter unexpectedly” v dmesg
  - Občas způsobily kernel panic
- Navenek projev jako drobný packet-loss

Co se vlastně stalo?

Restarty sítě:

- Odpojení lokálního Consul agenta od clusteru
  - Patroni se nemohly připojit a zjistit stav clusteru
  - Velmi častá změna mastera <- Patroni fungovaly dobře, což paradoxně způsobilo problémy.

Jak to opravit?

- Zapnout dočasně maintenance mode
  - Patroni přestaly řídit postgres a ten chvilkové síťové výpadky v pohodě zvládl
- Vypnout offloading
  - Tcp-segmentation-offload
  - Generic-segmentation-offload
  - generic-receive-offload
  - `ethtool -K eth0 gso off gro off tso off`
  - To okamžitě napravilo situaci

Rozbilo to index!

- Zaseklé jednoduché SELECTY
- Rozbitá data nebo rozbitý index?

https://pngtree.com/freepng/the-owl-staring-in-front_3324955.html.jpg

```
user_lists=# reindex database user_lists;
ERROR:  could not create unique index
"index_user_list_items_on_user_list_id_and_asset_id"
DETAIL:  Key (user_list_id, asset_id)=(ab22fcff-0d5f-476b-902c-3c29d3221228,
cfd2b50e-152d-2278-88db-a72b4b1bf22d) is duplicated.
```

- Duplicity i přes unique index

Postřehy a rady

- V případě použití consulu
  - Nechte Patroni připojovat přímo do hlavního clusteru
- Náš setup consulu je trochu specifický
  - Hlavní cluster
  - Lokální client na serveru
  - Patroni se připojují k lokálnímu klientovi
- Zkontrolovat kvalitu a výkon síťové karty

Postřehy a rady

- Maintenance mode je váš kamarád
  - `patronictl pause` / `patronictl resume`
  - Patroni se přestanou starat o PostgreSQL pod nimi
  - Možno dělat manualní zásahy v Postgresu, nebo upgradovat patroni, atd
  - Pomohlo nám to stabilizovat situaci

Postřehy a rady

- Některé parametry jsou postgresu předávány jako cmdline argumenty pro `pgctl start`
  - Důležité např. `max_connections` nebo `wal_keep_segments`
  - společné pro celý cluster
  - patroni.yml - uvést v sekci `bootstrap::dcs::postgresql`
  - https://patroni.readthedocs.io/en/latest/dynamic_configuration.html

Postřehy a rady

- Správné nastavení timeoutů na HAProxy

Shrnutí

- Patroni jsou super!
- Prostě to funguje

https://forum.webflow.com/t/hurray-for-ix-2-0-beta/48446

- Nic není dokonalé a ani Patroni nejsou nerozbitné
  - Musí tomu ale přát okolnosti
- Nepodceňte správnost nastavení a testování :)

Přijďte to řešit s námi!

Hledáme další kolegy

tech.showmax.com

Děkuji za pozornost

Otázky?