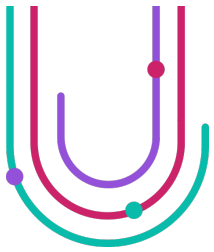




# Migration RepMgr vers Patroni

PGSessions 16  
Pierre BONHOMME  
07 février 2024





## Qui suis-je

Pierre  
BONHOMME

DBA depuis  
20ans

Multi Techno  
(SQL Server |  
Oracle | MySQL  
| PostGreSQL)

LinkedIn

## Qui sommes-nous

Ecritel

Hébergeur Web  
| SI  
Infogéreur

HADS & PCI-  
DSS | ISAE3402  
| ISO9001 |  
ISO27001

+200  
collaborateurs



Management  
de la sécurité



RGPD  
Infogérance & accompagnement



Management  
de la qualité



Hébergement & infogérance  
des données de santé



Norme de sécurité sur les données  
et transactions bancaires



Norme internationale de  
fiabilité du contrôle interne  
du service délivré

# SOMMAIRE

**Quésaco**

**Architecture actuelle**

**Architecture cible**

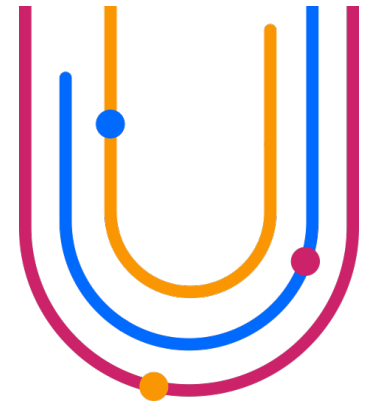
**Intérêts**

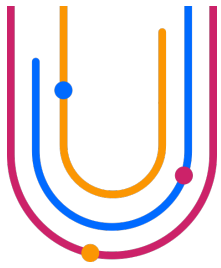
**Mode opératoire**

**Opérations**

**Conclusion**

**Questions**





**Quesaco ?**



**HA Proxy**

Solution d'équilibrage de charge haute disponibilité



**ETCD**

Magasin clé-valeur distribué



**RepMgr**

Outil de gestion de réplication à destination d'instances PostgreSQL

**Nagios**

**Nagios**

Solution de supervision



**Patroni**

Outil de gestion de réplication à destination d'instances PostgreSQL



**Haute  
Disponibilité**  
*Capacité d'un  
système à être  
disponible tout le  
temps*





## Architecture actuelle



### Produits engagés

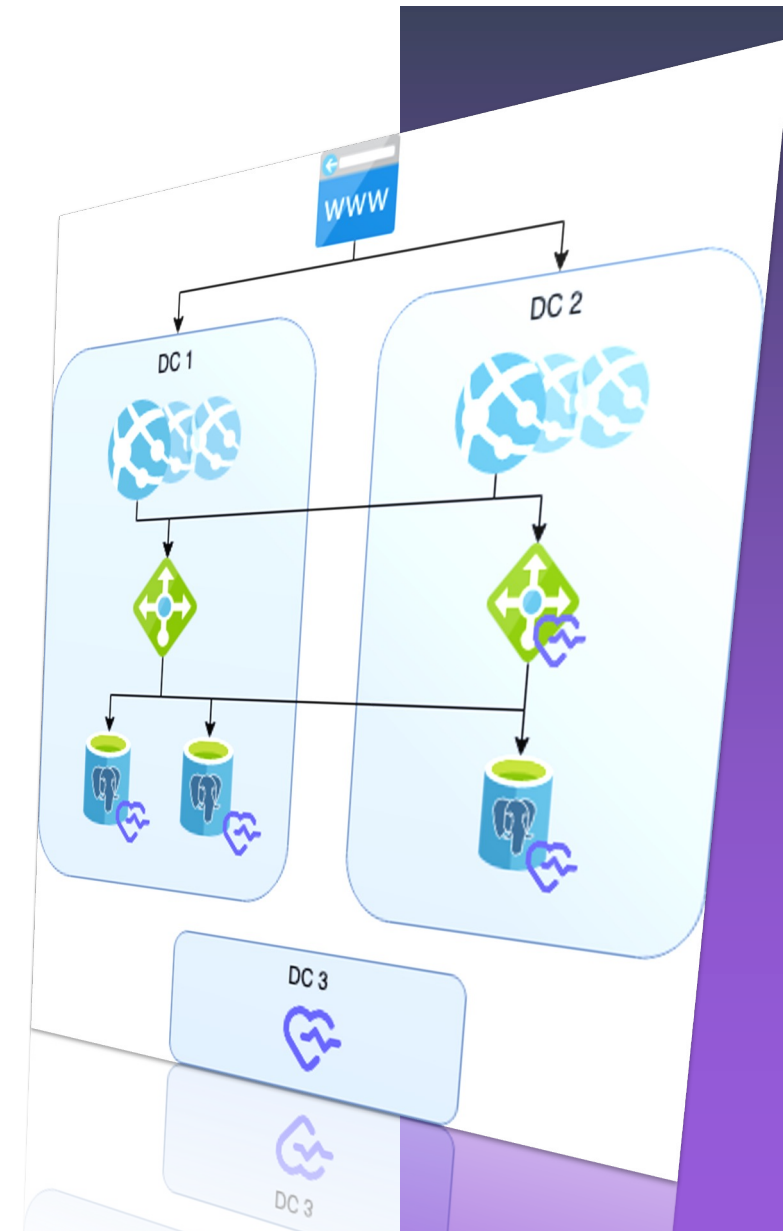
- Keepalive
- HaProxy
- RepMgr
- PostGreSQL

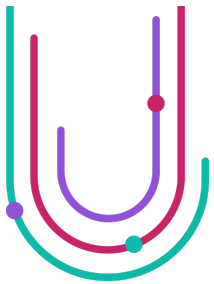
### Répartition sur 3 DC

- 5 Nœuds votant dans le cluster RepMgr
- 3 Nœuds participant à la réplication
- 2 nœuds en tant que témoins

### Haute Disponibilité / Bascule auto

- Des connexions via HAProxy
- Des rôles grâce à RepMgr





## Architecture Cible

### ■ Produits engagés

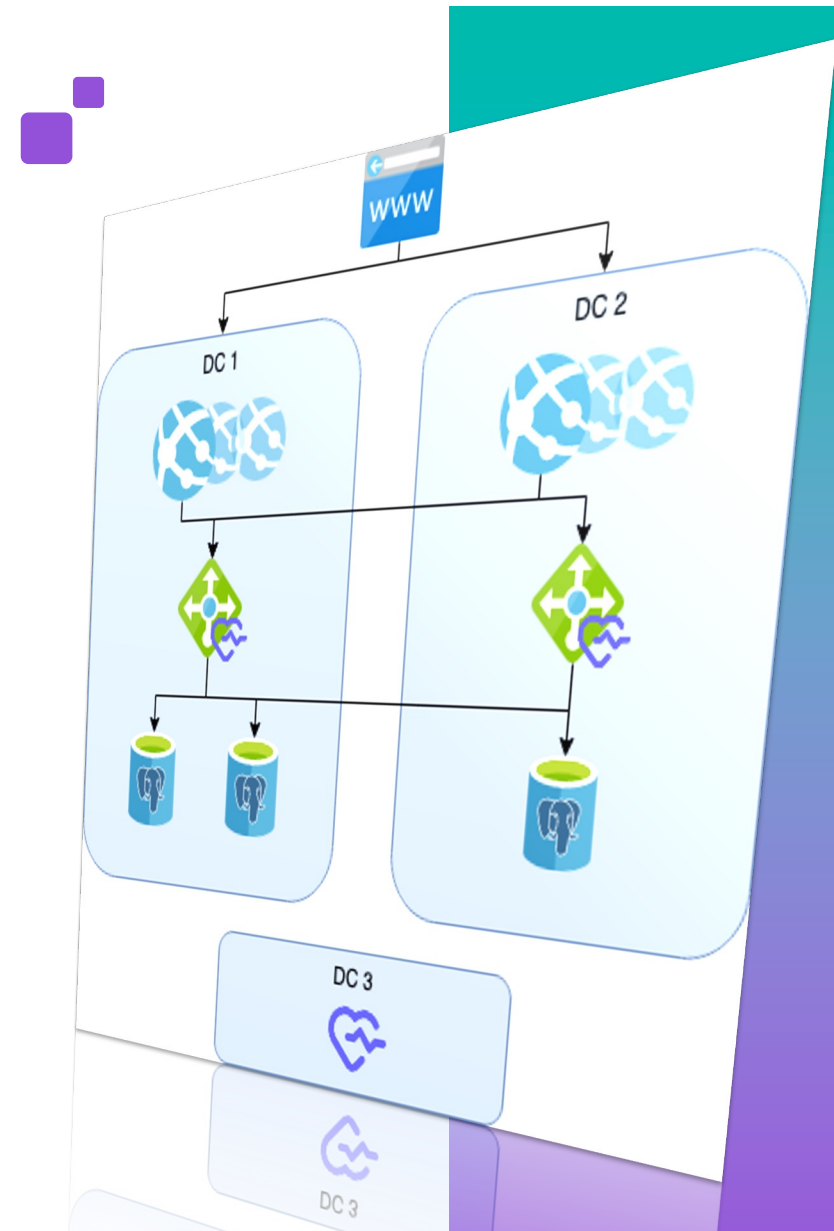
- Keepalive
- HaProxy
- etcd
- Patroni
- PostgreSQL

### ■ Répartition sur 3 DC

- 3 Nœuds ETCD
- 3 Nœuds participant à la réplication PostgreSQL

### ■ Bascule automatique

- Des connexions via HAProxy
- Des rôles grâce à Patroni





## Intérêts



### Accessibilité

L'exploitation de Patroni est facilitée par des commandes d'administration simples



### Licence

Solution Open-Source



### Légèreté

Déploiement des ETCD sur des machines externes avec des ressources contenues



### Pérennité

Solution Open-Source ne dépendant pas d'un éditeur



### Formation

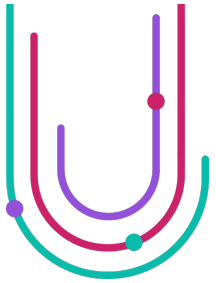
Possibilité de former les collaborateurs.trices via Dalibo HAPat



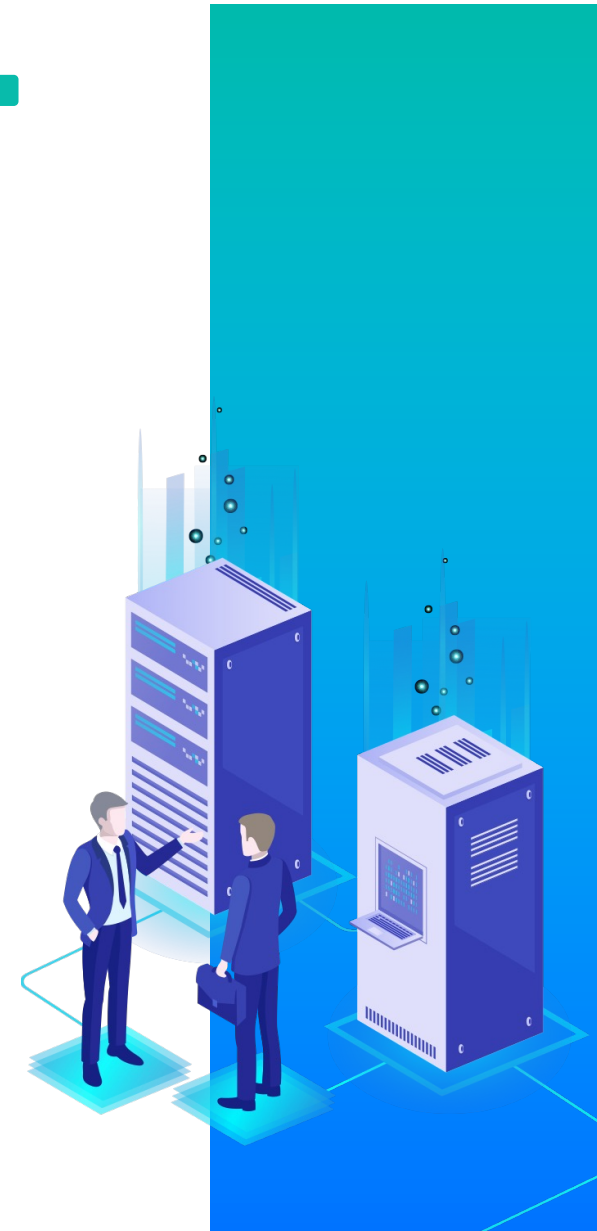
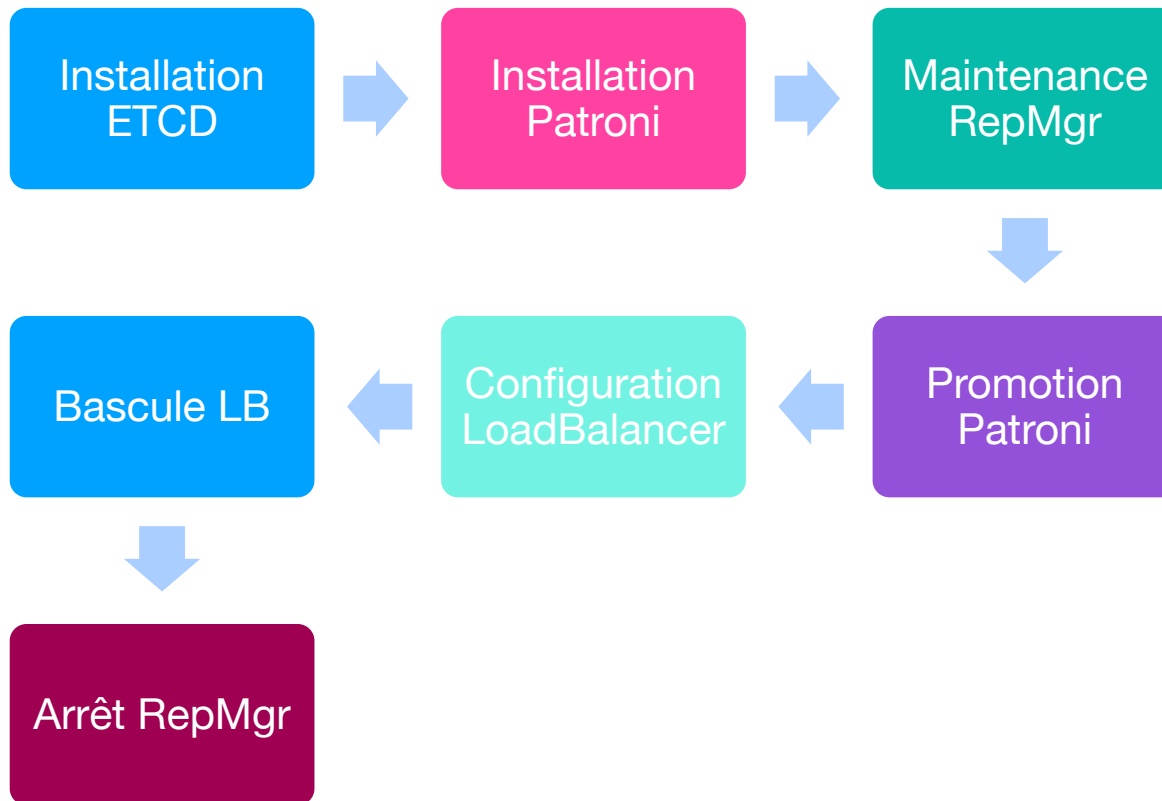
### Support

Communauté plus active

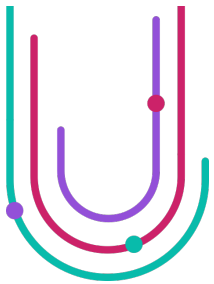




## Mode opératoire







## Opération : Installation ETCD

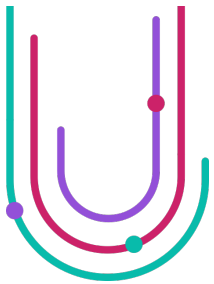


Via le script de Benoit **LOBREAU**

```
root@pocpat-lbpg-01:~# cat /etc/default/etcd
#[Member]
ETCD_DATA_DIR="/var/lib/etcd/default.etcd"
ETCD_LISTEN_PEER_URLS="http://172.16.69.2:2380"
ETCD_LISTEN_CLIENT_URLS="http://172.16.69.2:2379,http://127.0.0.1:2379,http://[:1]:2379"
ETCD_NAME="pocpat-lbpg-01"

#[Clustering]
ETCD_INITIAL_ADVERTISE_PEER_URLS="http://172.16.69.2:2380"
ETCD_ADVERTISE_CLIENT_URLS="http://172.16.69.2:2379"
ETCD_INITIAL_CLUSTER="pocpat-lbpg-01=http://172.16.69.2:2380,pocpat-lbpg-02=http://172.16.69.3:2380,pocpat-witness-01=http://172.16.69.4:2380"
ETCD_INITIAL_CLUSTER_TOKEN="patroni-clusters"
ETCD_INITIAL_CLUSTER_STATE="new"

#[Other stuff]
ETCD_ENABLE_V2="true"
```

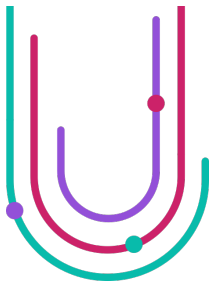


## Opération : Installation ETCD



Via le script de Benoit **LOBREAU**

```
root@pocpat-lbpg-02:~# etcdctl -w table endpoint status
+-----+-----+-----+-----+-----+-----+-----+
-+
|      ENDPOINT      |      ID      | VERSION | DB SIZE | IS LEADER | RAFT TERM | RAFT INDEX |
|-----|-----|-----|-----|-----|-----|-----|
-+
| http://172.16.69.2:2379 | 8abb88d674866e40 | 3.3.25 | 20 kB | false | 3 | 1731 |
| http://172.16.69.3:2379 | 2393976ac18ba645 | 3.3.25 | 20 kB | false | 3 | 1731 |
| http://172.16.69.4:2379 | 10b659bddb70a65e | 3.3.25 | 20 kB | true | 3 | 1731 |
|-----|-----|-----|-----|-----|-----|
-+
```



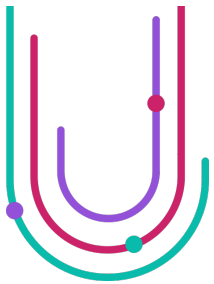
## Opération : Installation Patroni



Via le script de Benoit **LOBREAU**

```
scope: cluster-main-13
namespace: /service/ # valeur par défaut
name: pocpat-db-01

restapi:
  listen: 172.16.69.5:8008
  connect_address: 172.16.69.5:8008
log:
  level: INFO
  dir: /var/log/postgresql/
etcd:
  hosts:
    - 172.16.69.2:2379
    - 172.16.69.3:2379
    - 172.16.69.4:2379
  username: patroni
  password: patroni
  protocol: http
bootstrap:
```



# Opération : Installation Patroni



Documentation « Convert a Standalone to a Patroni Cluster »  
[https://patroni.readthedocs.io/en/latest/existing\\_data.html](https://patroni.readthedocs.io/en/latest/existing_data.html)

## Création des utilisateurs

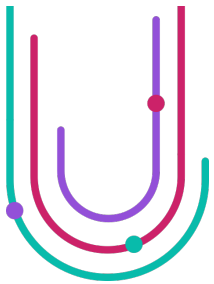
- patroni super utilisateur
- patroni gestionnaire de la réplication
- pg\_rewind

## Paramétrage de Patroni

- Transformation du fichier de configuration de PostgreSQL en yaml (= > :)
- `patroni --validate-config /etc/patroni/config.yml`

## Désactiver le service postgresql

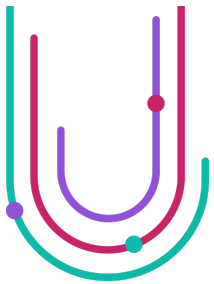
- L'instance PostgreSQL étant géré par patroni, il faut désactiver le service dans le système
- `systemctl disable postgresql@13-main.service`



## Opération : Maintenance RepMgr



```
postgres@pocpat-db-01:~$ repmgr -f /etc/postgresql/13/main/repmgr.conf service status
ID | Name          | Role      | Status      | Upstream      | repmgrd      | PID | Paused? | Upstream
last seen
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----
3  | pocpat-lbpg-02 | witness  | * running  | pocpat-db-01 | not running  | n/a | n/a     | n/a
4  | pocpat-witness-01 | witness | * running  | pocpat-db-01 | not running  | n/a | n/a     | n/a
5  | pocpat-db-01   | primary  | * running  |               | not running  | n/a | n/a     | n/a
6  | pocpat-db-02   | standby  | running    | pocpat-db-01 | not running  | n/a | n/a     | n/a
7  | pocpat-db-03   | standby  | running    | pocpat-db-01 | not running  | n/a | n/a     | n/a
postgres@pocpat-db-01:~$ repmgr -f /etc/postgresql/13/main/repmgr.conf service pause
NOTICE: node 3 (pocpat-lbpg-02) paused
NOTICE: node 4 (pocpat-witness-01) paused
NOTICE: node 5 (pocpat-db-01) paused
NOTICE: node 6 (pocpat-db-02) paused
NOTICE: node 7 (pocpat-db-03) paused
```

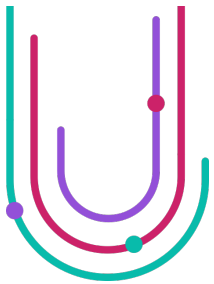


## Opération : Promotion Patroni



```
root@pocpat-db-01:~# systemctl start patroni
root@pocpat-db-01:~# systemctl status patroni
• patroni.service - Runners to orchestrate a high-availability PostgreSQL
   Loaded: loaded (/lib/systemd/system/patroni.service; enabled; vendor preset: enabled)
   Active: active (running) since Wed 2024-01-10 12:40:03 CET; 1min 17s ago
 Main PID: 69349 (patroni)
   Tasks: 5 (limit: 4657)
  Memory: 24.9M
     CPU: 344ms
   CGroup: /system.slice/patroni.service
           └─69349 /usr/bin/python3 /usr/bin/patroni /etc/patroni/config.yml

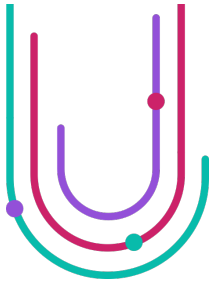
janv. 10 12:40:03 pocpat-db-01 systemd[1]: Started Runners to orchestrate a high-availability PostgreSQL.
root@pocpat-db-01:~# patronictl -c /etc/patroni/config.yml list
+ Cluster: main (7317665090420293452) +-----+-----+-----+
| Member          | Host          | Role   | State   | TL | Lag in MB |
+-----+-----+-----+-----+-----+-----+
| pocpat-db-01   | 172.16.69.5  | Leader | running | 3  |           |
+-----+-----+-----+-----+-----+-----+
```



## Opération : Configuration LB



```
root@pocpat-lbpg-01:~# tail -14 /etc/haproxy/haproxy.cfg
listen frontend_pg
  mode tcp
  balance leastconn
  stats enable
  stats uri /
  bind *:5432
  option httpchk OPTIONS /master
  http-check expect status 200
  fullconn 320
  default-server inter 3s fall 3
# Servers
server pocpat-db-01 172.16.69.5:5432 check port 8008 maxconn 320 check inter 10s
server pocpat-db-02 172.16.69.6:5432 check port 8008 maxconn 320 check inter 10s
server pocpat-db-03 172.16.69.7:5432 check port 8008 maxconn 320 check inter 10s
root@pocpat-lbpg-01:~# haproxy -c -f /etc/haproxy/haproxy.cfg
Configuration file is valid
```



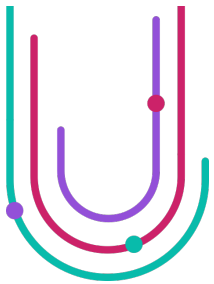
## Opération : Bascule LB



```
root@pocpat-lbpg-01:~# systemctl restart haproxy
root@pocpat-lbpg-01:~# systemctl status keepalived

root@pocpat-lbpg-02:~# systemctl stop keepalived
root@pocpat-lbpg-02:~# systemctl start keepalived
root@pocpat-lbpg-02:~# systemctl restart haproxy
```





## Opération : Arrêt RepMgr



### Arrêt RepManager

- `systemctl stop repmgrd.service`
- `systemctl disable repmgrd.service`

### Désinstallation

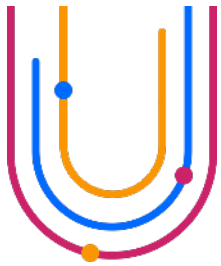
- `apt remove repmgr`

### Nettoyage

- Suppression anciennes lignes dans haproxy
- Purge fichier de configuration repmgr
- Suppression utilisateur repmgr

### Et Après

- Planifier un test de PCA !



## Conclusion



Patroni est la solution idéale pour ce type d'architecture :

- Facilement implémentable avec un LoadBalancer HAProxy via les API intégrés
- Léger
- Facile à prendre en main

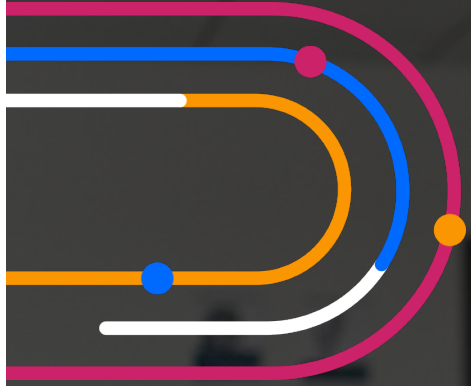
Dans une migration comme celle-ci, le plus compliqué est de faire cohabiter les deux technologies durant la migration sans impacter les services satellites.

Entre la possibilité de suspendre la gestion du cluster dans RepMgr et la possibilité que Patroni soit transparent jusqu'à la prochaine relance des nœuds, les plus grosses difficultés sont le maintien des fichiers de configurations (postgresql => patroni).

La coupure est infime via la bascule de la VIP. HAProxy permet même le luxe de pouvoir recharger sa configuration sans perte de paquet (non appliqué ici)

*“ Coupure minimale  
Client satisfait ”*





Question ?

