

# Architecture de l'application MXT

# Organisation des services

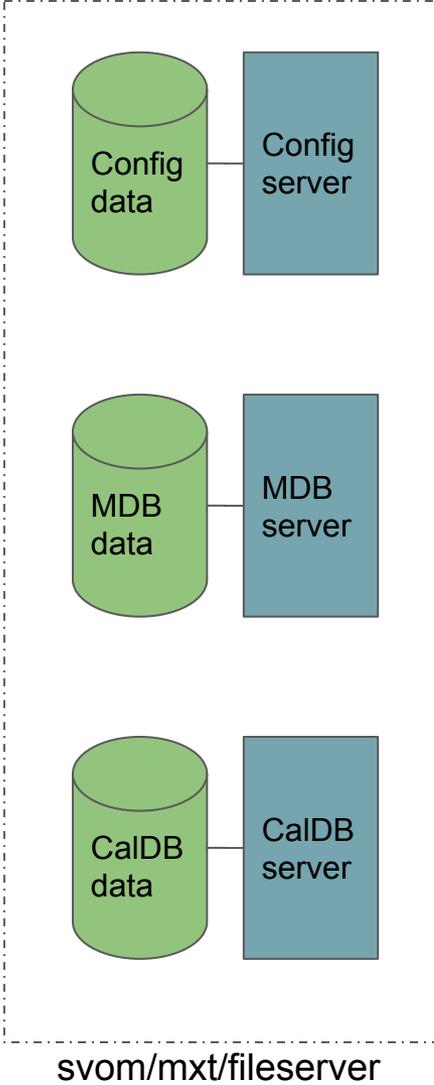
Persistance

Services

Frontend

# Organisation des services

## Persistence

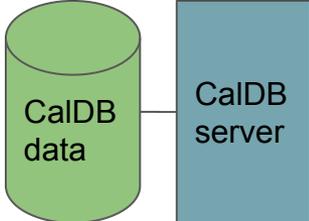
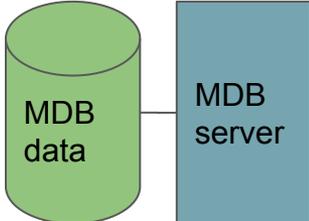
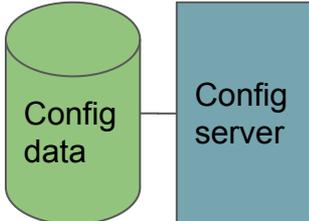


## Services

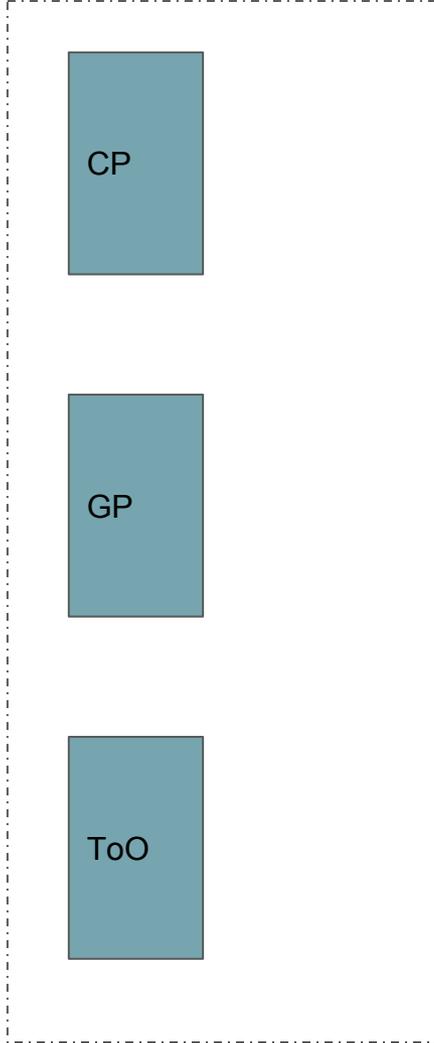
## Frontend

# Organisation des services

## Persistence



## Services

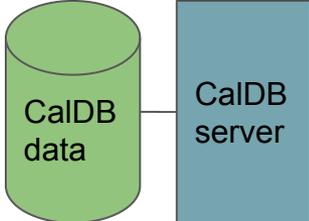
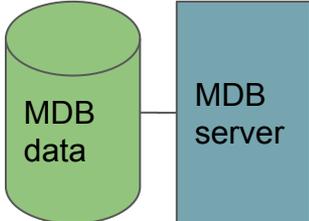
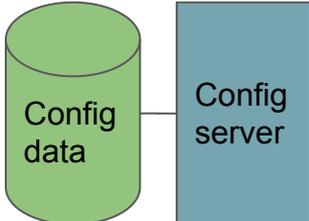


pipeline-bricks

## Frontend

# Organisation des services

## Persistence



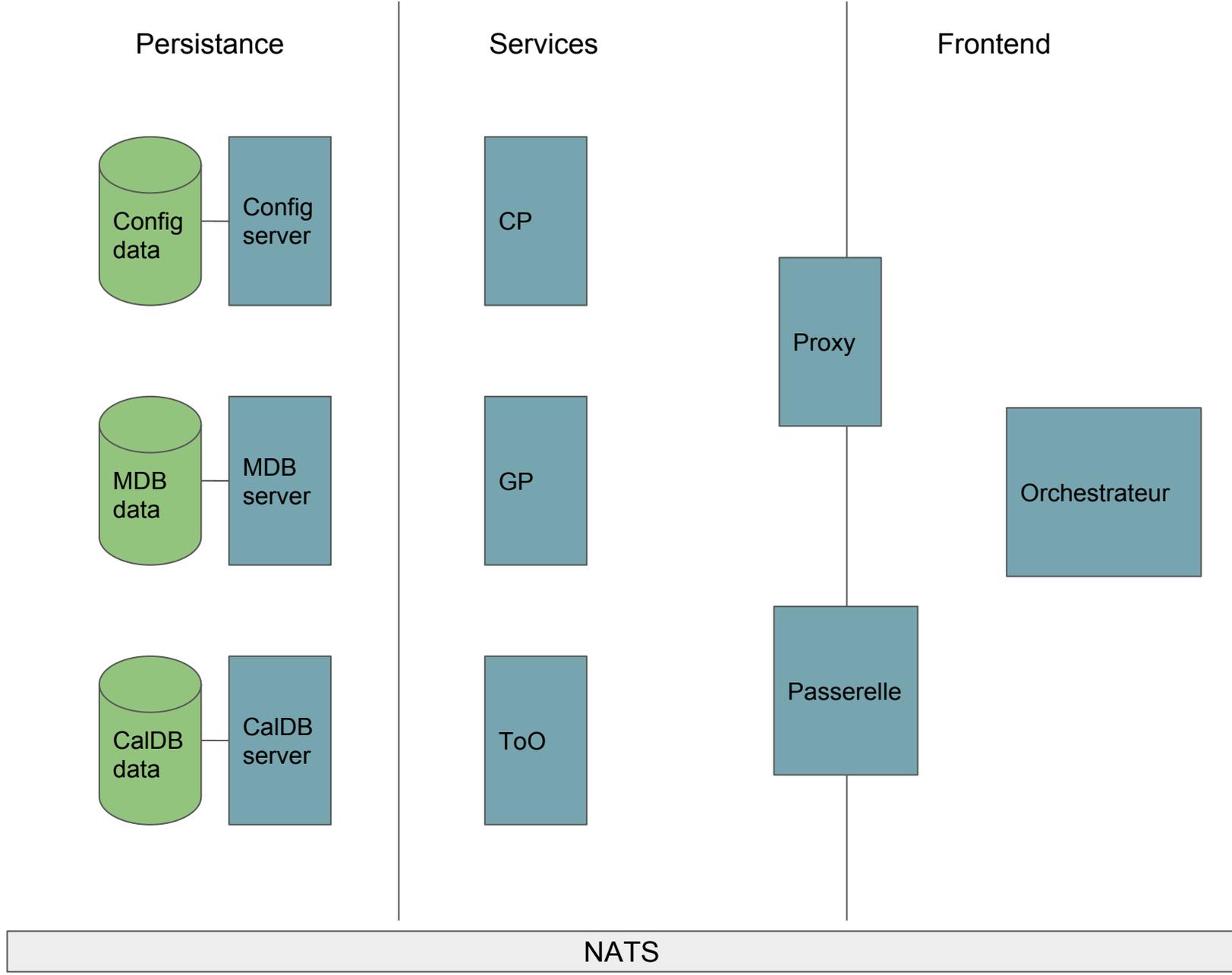
## Services



## Frontend



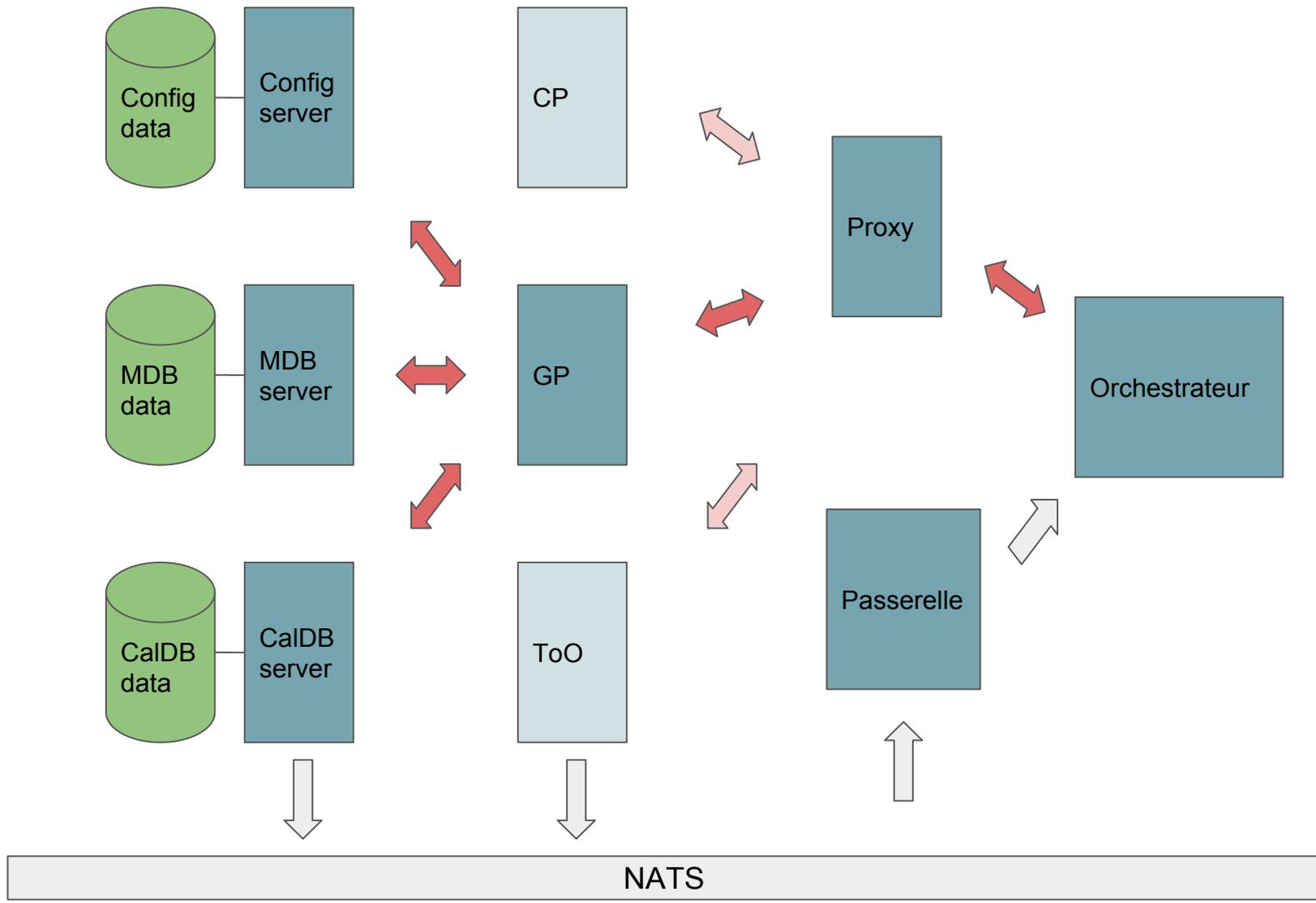
# Communication entre les services



# Communication entre les services

➔ NATS

↔ REST



# Conclusion

- Architecture fonctionnelle
  - Lancer des traitements
  - Sauvegarder les données
  - Récupérer les logs
- Permet de gérer les traitements indépendamment de l'architecture
- Demande une intégration à l'architecture globale
- Reste à intégrer la physique



# Quelques mots sur le Proxy

Utilisation de Nginx en mode **reverse proxy** -> projet [svom/mxt/proxy](https://svom.mxt/proxy)

Objectif:

- Minimiser les ports ouverts vers l'extérieur

Avantages:

- Un seul point d'entrée
- Adresses plus simples
- Déploiement facilité
- Plus de sécurité

proxy path	service	adresse
/mxt/	orchestrator	<a href="http://master:1024/">http://master:1024/</a>
/mxt_configdb/	configdb	<a href="http://config:1024/">http://config:1024/</a>
/mxt_missiondb/	mission db	<a href="http://missiondb:1024/">http://missiondb:1024/</a>
/caldb/	CALDB	<a href="http://caldb:1024/">http://caldb:1024/</a>
/mxt_gp_pipeline/	General program	<a href="http://generalprogram:1024/">http://generalprogram:1024/</a>
/mxt_cp_pipeline/	Core program	<a href="http://coreprogram:1024/">http://coreprogram:1024/</a>
/mxt_too_pipeline/	Target of opportunity	<a href="http://too:1024/">http://too:1024/</a>
/mxt_websocket/	websocket	<a href="ws://gateway:1024/">ws://gateway:1024/</a>

# Quelques mots sur le Proxy

Difficultés :

- le proxy doit être démarré en dernier car il ne 'voit' pas les nouveaux services apparaître

Solution:

- Remettre à jour la config automatiquement
- [jwilder/nginx-proxy](#)
  - Surcouche a nginx
  - Connection au démon docker