



Migrer sa fonctionnalité la plus **critique** sans **régression** ni coupure de **service**

Florian MARIN  
Software Engineering Manager

**AZUR TECH**  
*Winter*

**MERCI**  
**à toutes et tous d'être venu**



Software Engineering Manager

Teads depuis 2020

Web Platform Team



florian-marin



flomarin88



flomarin

Des questions ? Envie de discuter ?  
Mes messages privés sont ouverts



# Agenda

## **Contexte**

Fonctionnalité critique qui ne supporte plus notre business

---

## **Résultats**

Satisfaisants après 2 ans de projet

---

## **Démarche**

Étapes clés de cette migration

---

## **Conclusion**

Quelques leçons apprises



# Disclaimer



# Contexte

# Point d'entrée de nos clients

Gérer leurs campagnes de publicités digitales en autonomie

Fonctionnalité **LA plus critique**

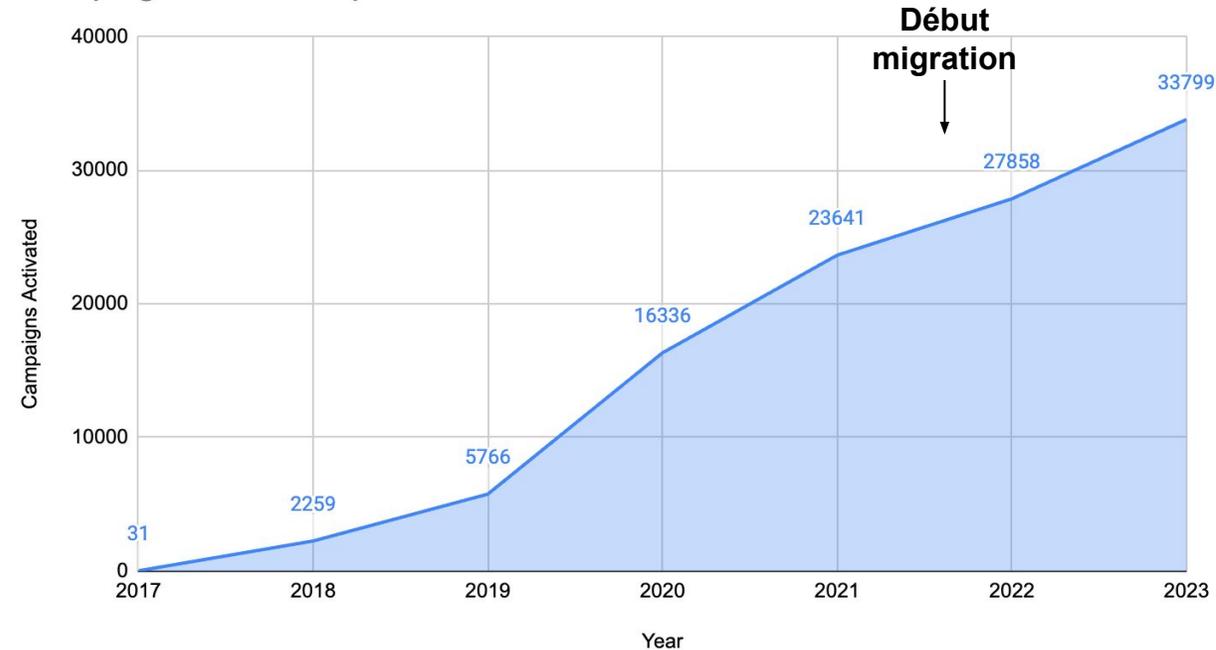
Disponibilité attendue : **24 / 7**

## Stratégie de Teads

D'un business model "Agence" vers "SaaS"

Application devient au centre de la stratégie

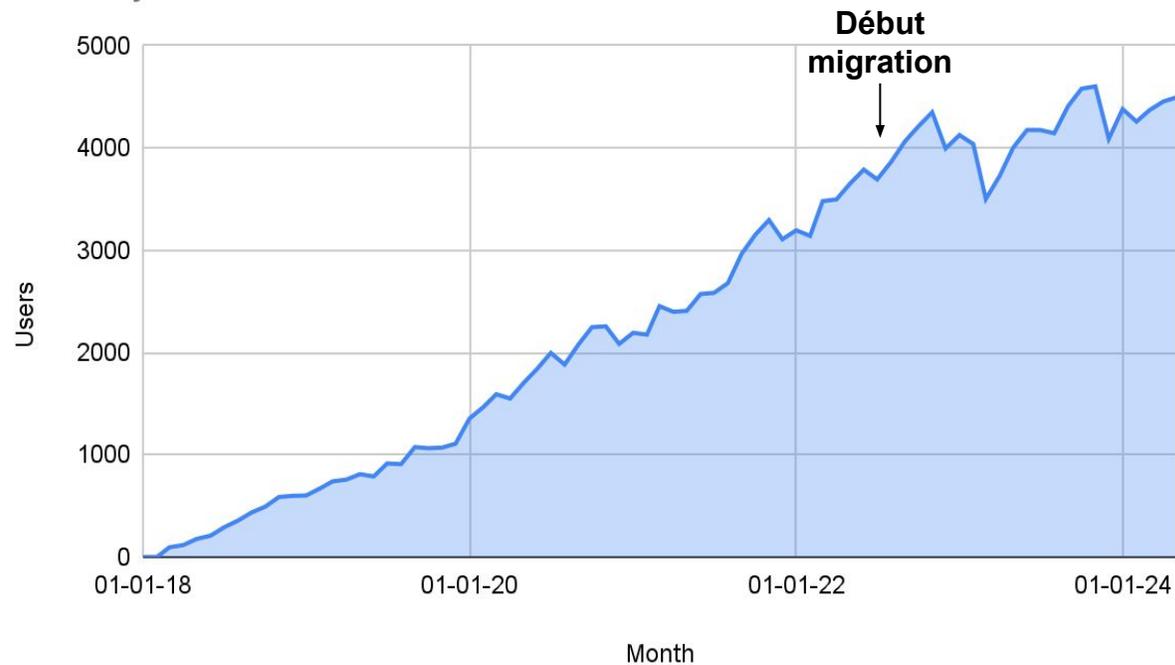
Campaigns Activated per Year



# Hausse du trafic qui stresse le système

Et surtout l'organisation, finalement pas prête pour cette nouvelle stratégie

Monthly active users



5%

Taux d'erreur

# Des contributeurs de plus en plus nombreux

Dans une base de code en constante évolution

Application initialement développée par une seule équipe, de A à Z.

Puis ownership s'est décentralisé

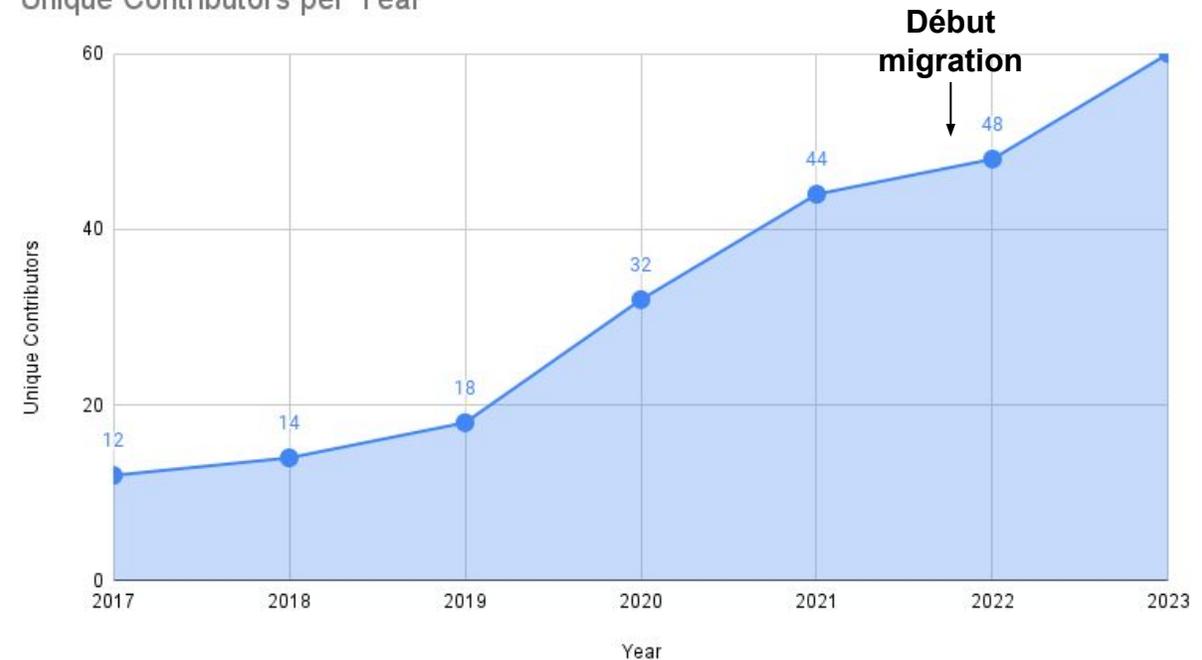
## Contributeurs

- 2017: 12 / 1 équipe
- Aujourd'hui: 60 / +10 équipes

## Plateforme : Création d'une équipe

- Fin 2021 : 3 personnes (2 Senior, 1 Junior)
- Aujourd'hui : 10 personnes

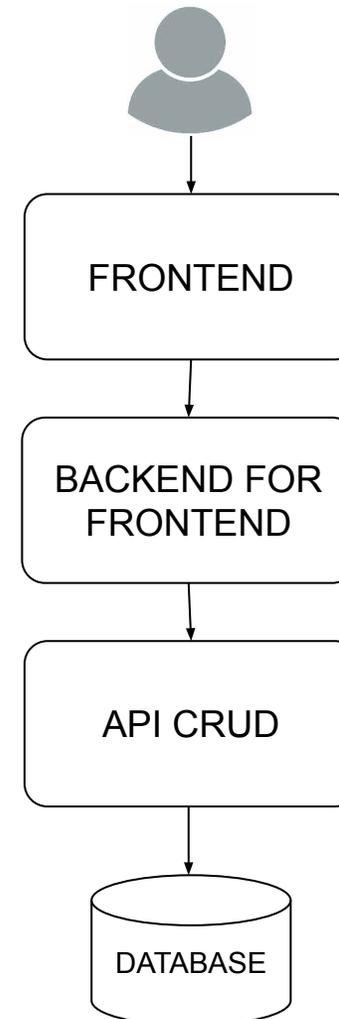
Unique Contributors per Year



# Une architecture à faire évoluer

En respectant les contraintes business et techniques de l'entreprise

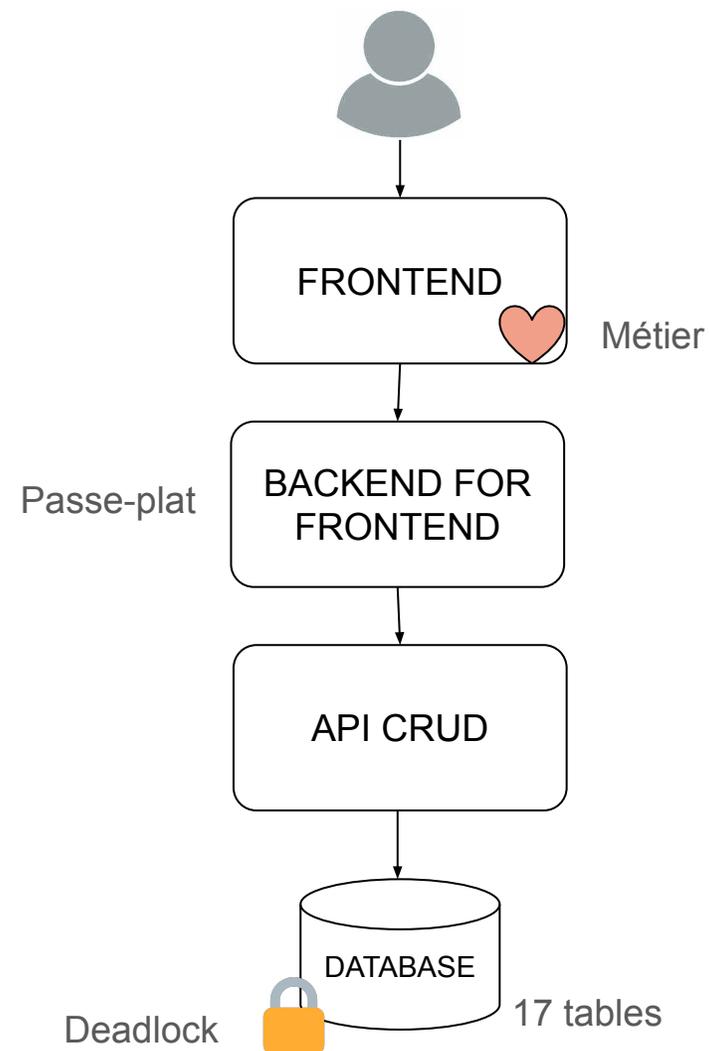
- Continuer à faire évoluer la fonctionnalité
- Conserver le même modèle de base de données
- Eviter les effets Tunnel et Big Bang



# Une architecture à faire évoluer

En respectant les contraintes business et techniques de l'entreprise

- Continuer à faire évoluer la fonctionnalité
- Conserver le même modèle de base de données
- Eviter les effets Tunnel et Big Bang

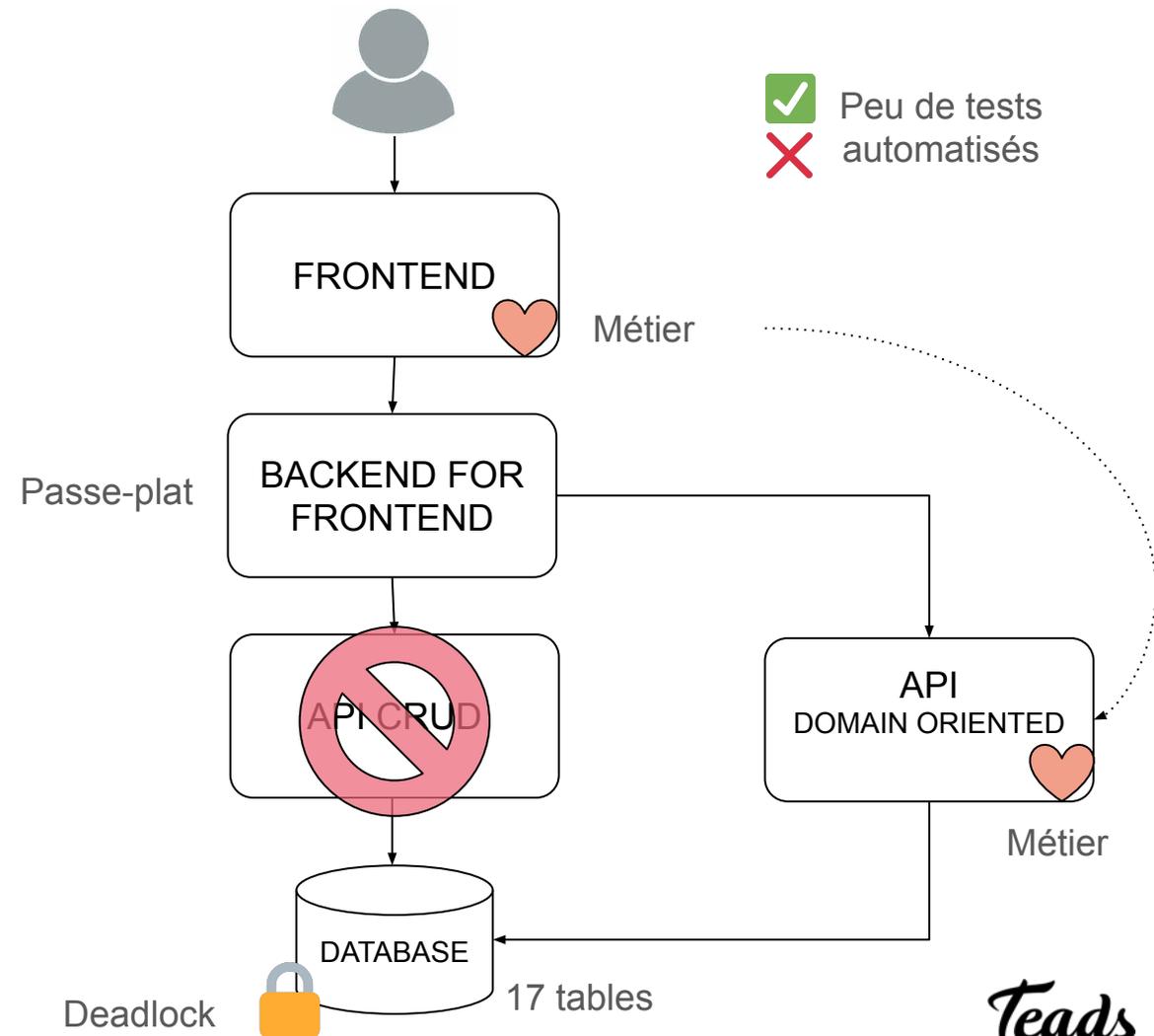


✓ Peu de tests automatisés  
✗

# Une architecture à faire évoluer

En respectant les contraintes business et techniques de l'entreprise

- Continuer à faire évoluer la fonctionnalité
- Conserver le même modèle de base de données
- Eviter les effets Tunnel et Big Bang





# Résultats

# Des améliorations significatives

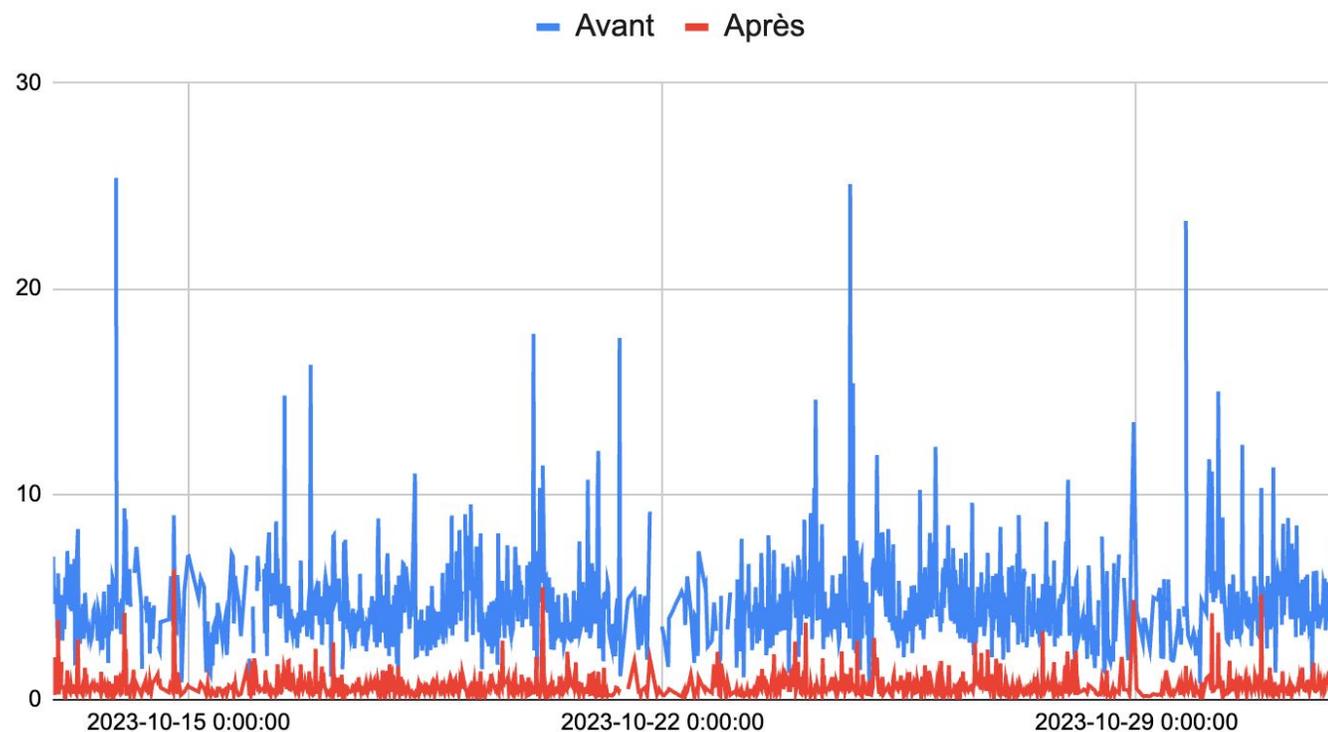
Et la migration n'est pas encore totalement terminée



0,15%

Taux d'erreur

Comparaison des latences (Avant / Après)





# Démarche

# Rédiger une vision de l'architecture finale

Avant même d'écrire une première ligne de code

## Design Document @Teads

- Contexte / Problème(s) à résoudre
- Solutions étudiées et sélectionnée avec argumentaires
- Impacts et dépendances
- Stratégie de tests
- Stratégie de déploiement
- Macro Planning
- Coûts

Validation inter équipes

Public => Engagement fort



# Monitorer lors toutes les phases

*"If you can't measure it, you can't improve it"*

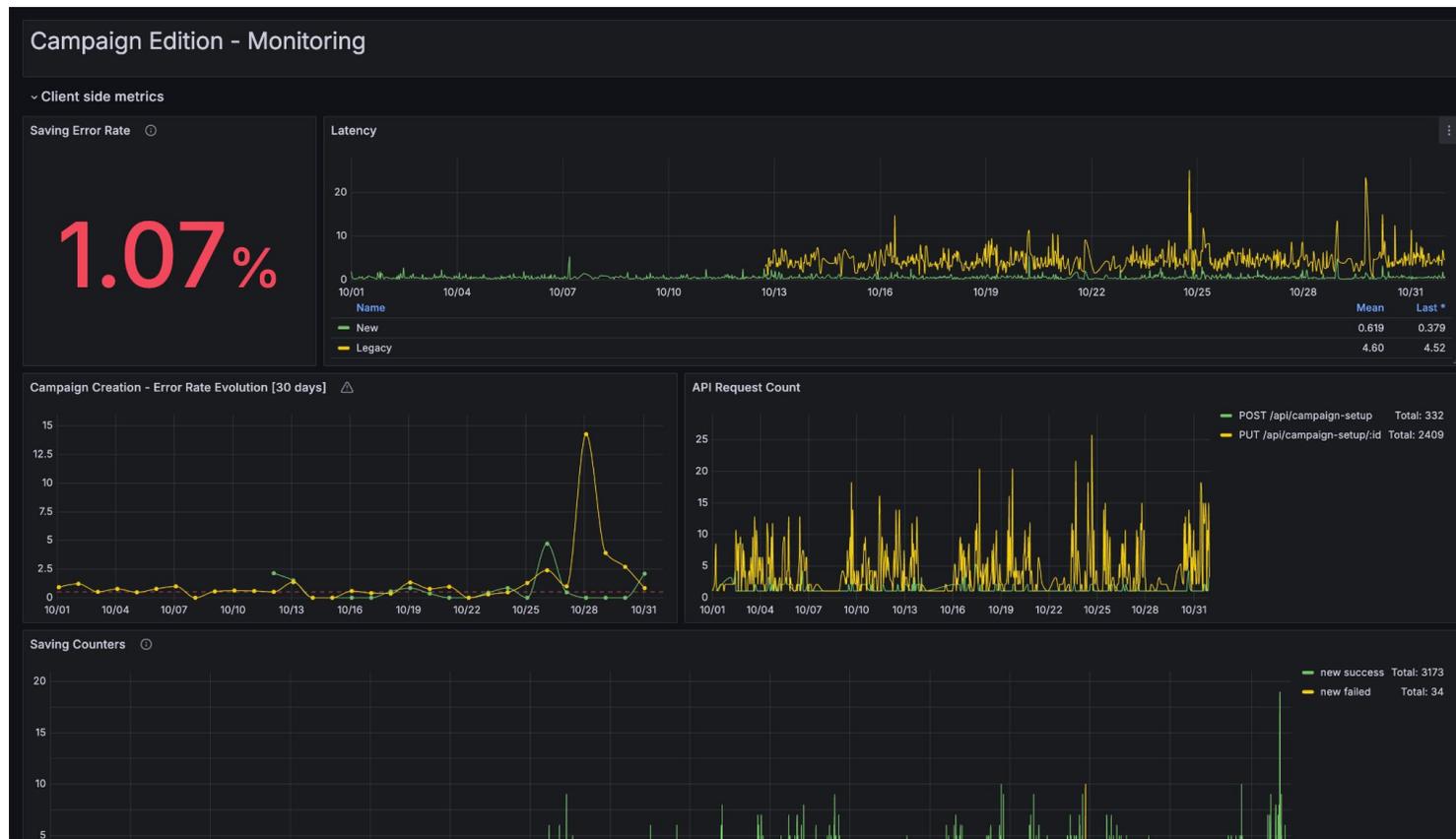
Dashboard spécifique de supervision

Metriques

- Taux d'erreur
- Latence
- Nombre d'appels API legacy

Alerte contextualisée à chaque erreur

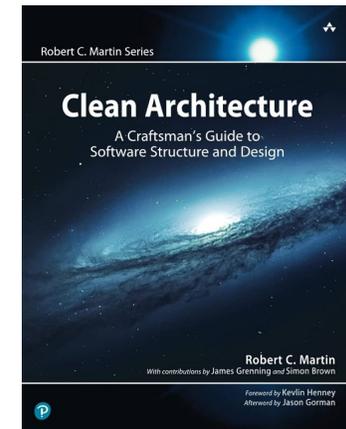
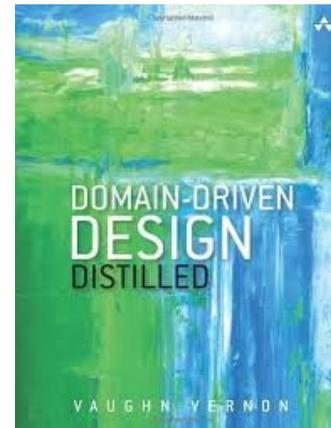
Outillez vous



# Définir un nouveau “domaine”

Supporté par une Clean Architecture

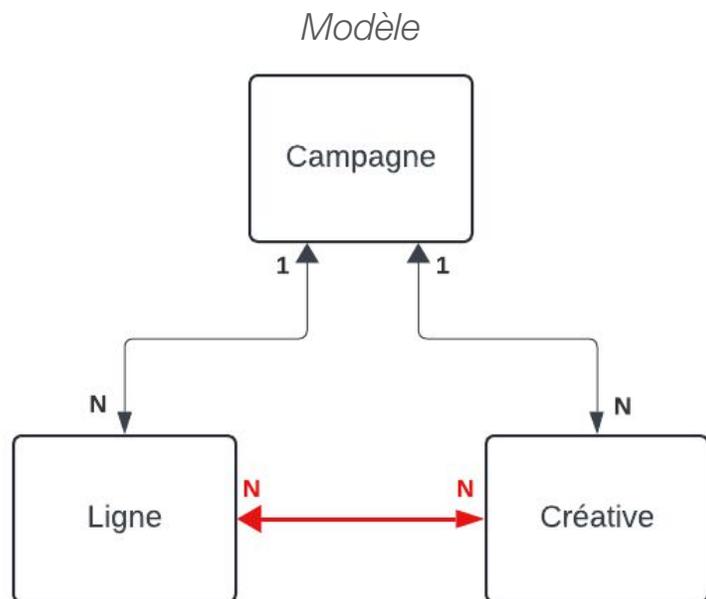
- Mettre le métier au coeur du code et de l'architecture
- Gérer entièrement le cycle de vie d'une configuration de campagne
- S'abstraire des couches d'infrastructure (DB) et de communication (GRPC)
- Définir un contrat API orienté “use case”



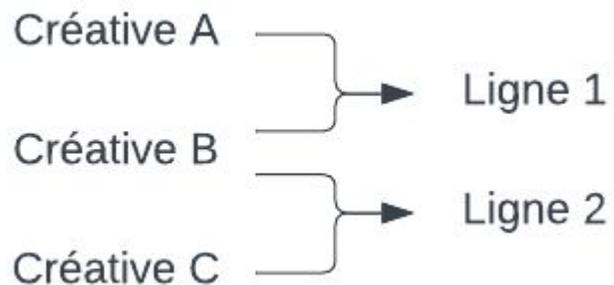
# Résoudre les problèmes un par un

Mode itératif qui permet de garder la motivation et de souffler de temps en temps

## Exemple



## Exemple simplifié



4 calls API en //

Chaque call fait une écriture dans la table "Ligne"

**Deadlock**

## Résolution

- **✗** Séquentiel: Impact les performances et si un call échoue ?
- **✓** Batch : Endpoint GRPC `AssignCreativeToLineItems`

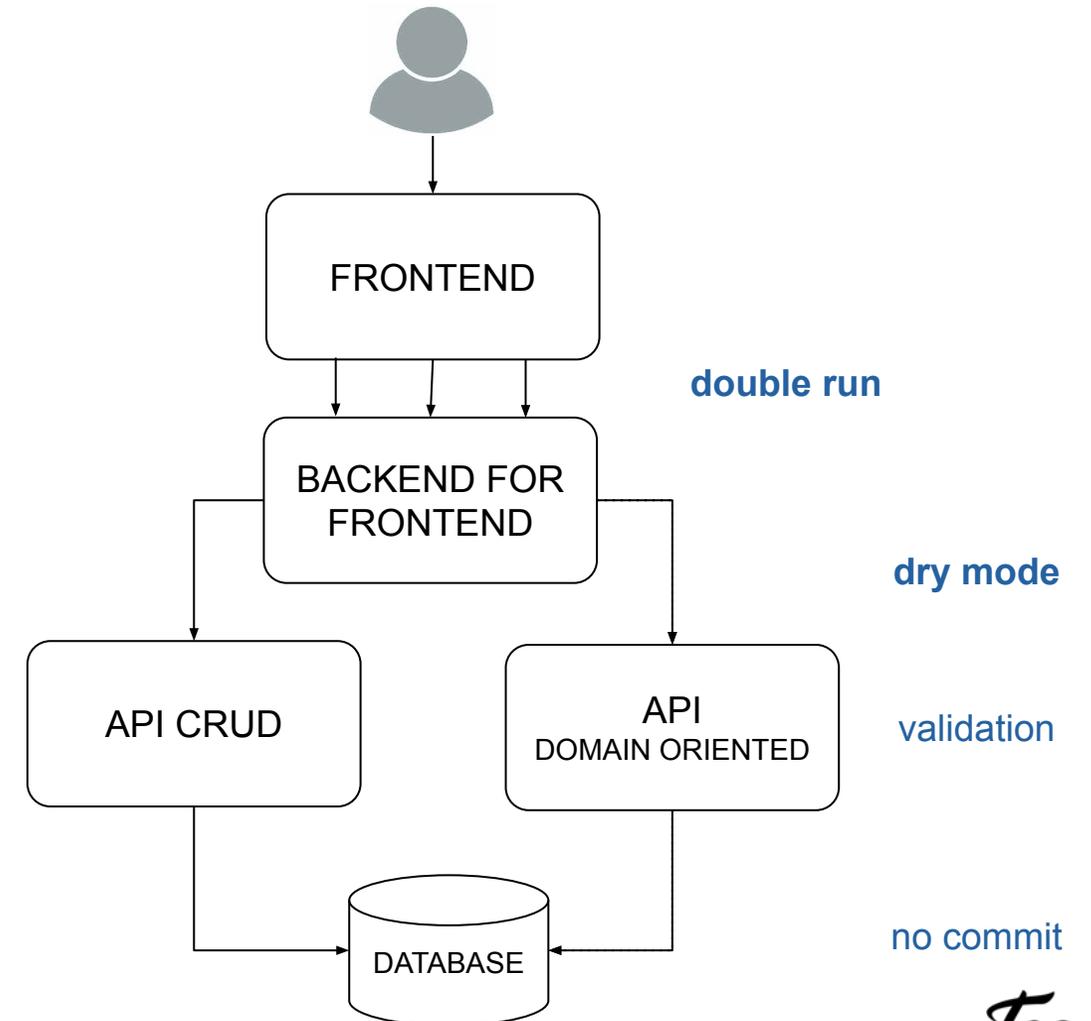
# Expérimenter l'architecture cible

Double Run en "dry" mode

Aucun impact pour l'utilisateur

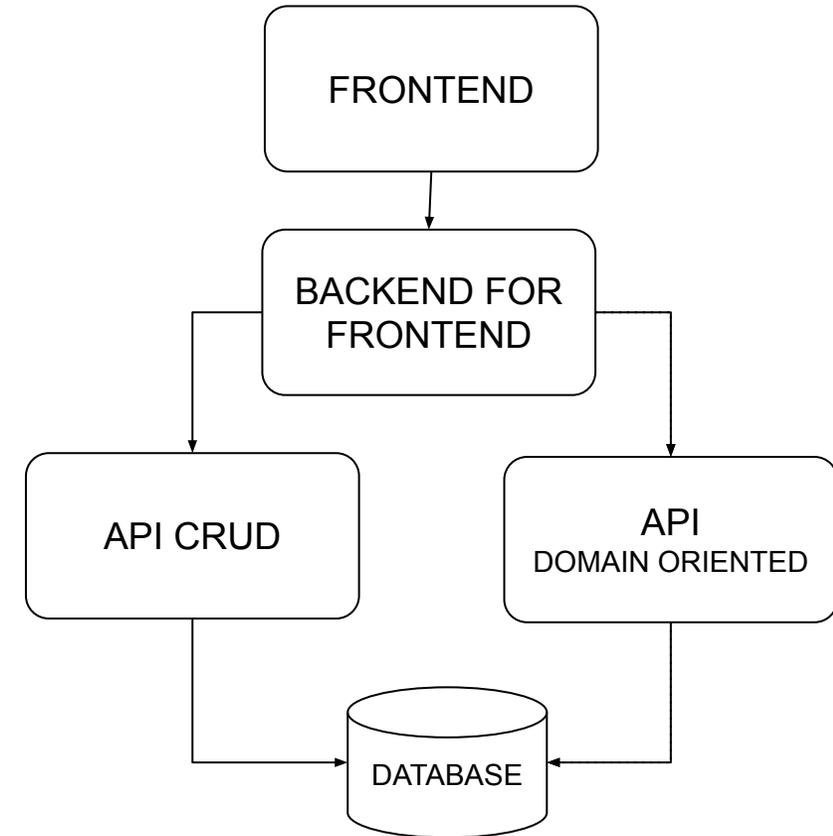
Prendre le temps d'explorer les uses cases

Monitorer les résultats



# Adopter une stratégie de tests complète

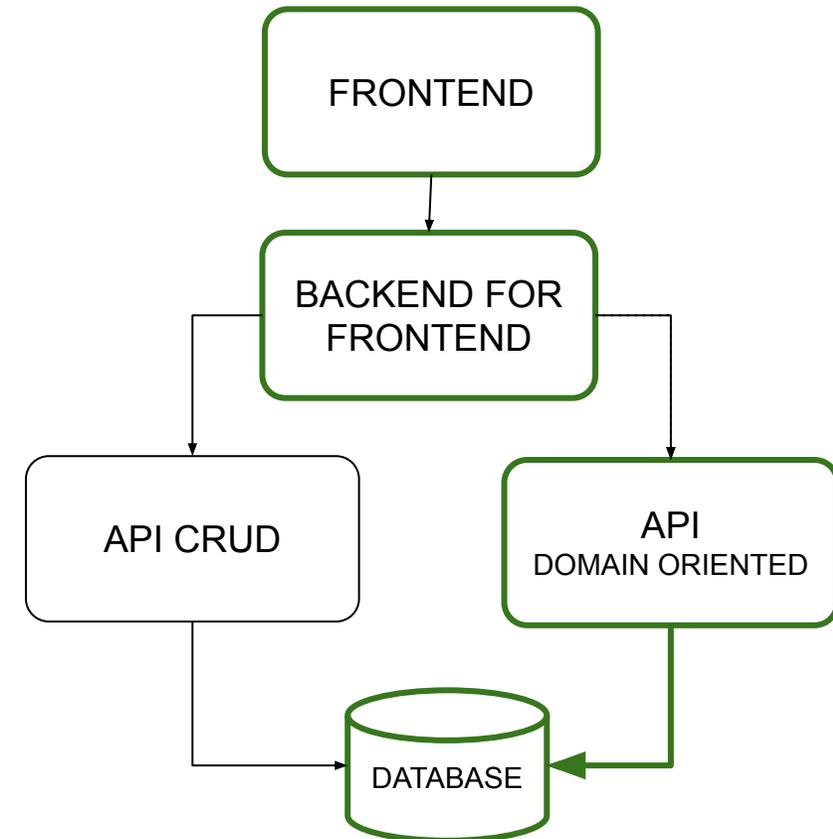
Récupérer de la connaissance métier et éviter les régressions



# Adopter une stratégie de tests complète

Récupérer de la connaissance métier et éviter les régressions

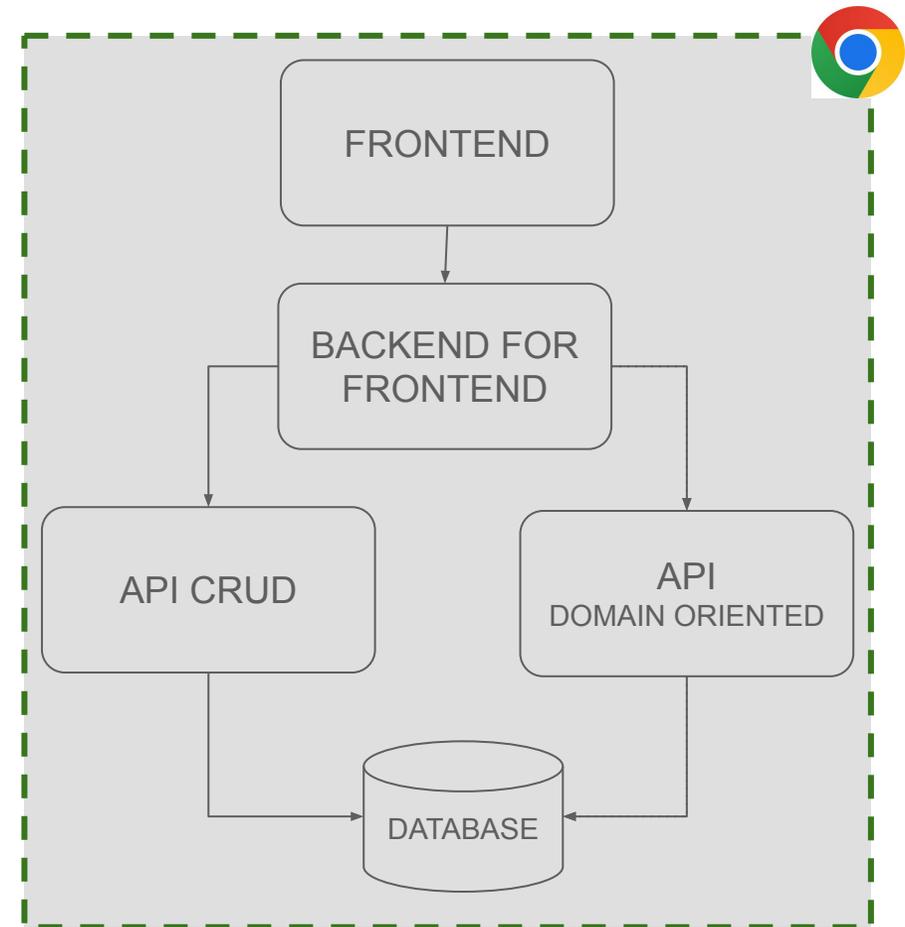
- Unitaires / Intégrations évidemment mais pas suffisants



# Adopter une stratégie de tests complète

Récupérer de la connaissance métier et éviter les régressions

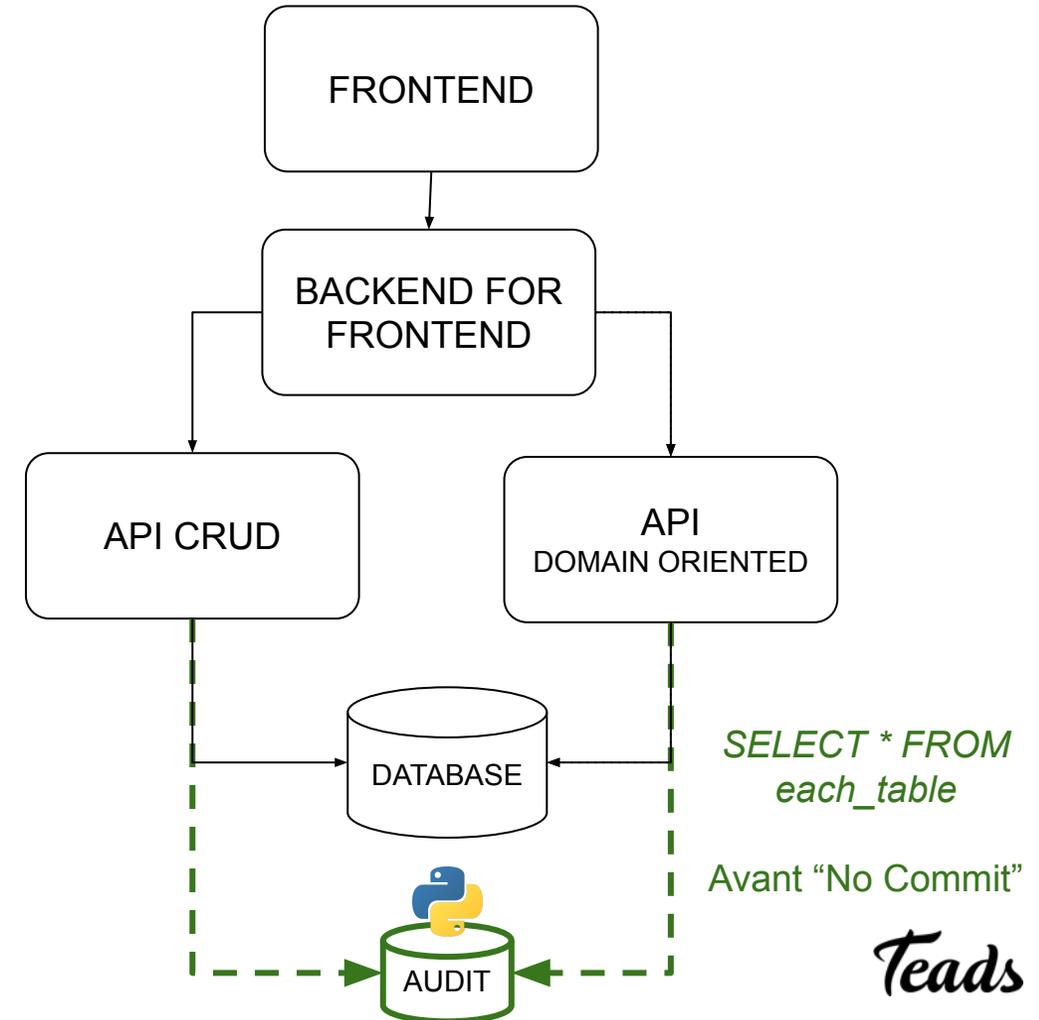
- Unitaires / Intégrations évidemment mais pas suffisants
- End to End dans la CI
  - Uniquement quelques scénario critiques



# Adopter une stratégie de tests complète

Récupérer de la connaissance métier et éviter les régressions

- Unitaires / Intégrations évidemment mais pas suffisants
- End to End dans la CI
  - Uniquement quelques scénario critiques
- **Production** directement 🤖
  - Comparaison quotidienne des *audits*

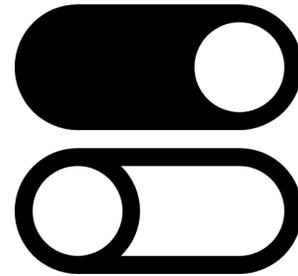


# Déployer progressivement

Un plan préparé, communiqué et à suivre à la lettre



**Communication**  
Interne et Clients



**Feature Flag**  
Ouverture hebdomadaire



**Suivi Incidents**  
Tout noter

# Conclusion

# Quelques leçons apprises

- Mesurer dès le début et en continu
- Définir la vision technique finale et se faire challenger par le reste des équipes
- Automatiser et diversifier les tests
- Déployer progressivement, en interne puis externe
- (Sur)Communiquer
- Anticiper en allant chercher les feedbacks et les informations sur les fonctionnalités à venir
- Soyez patients voire pessimistes parfois

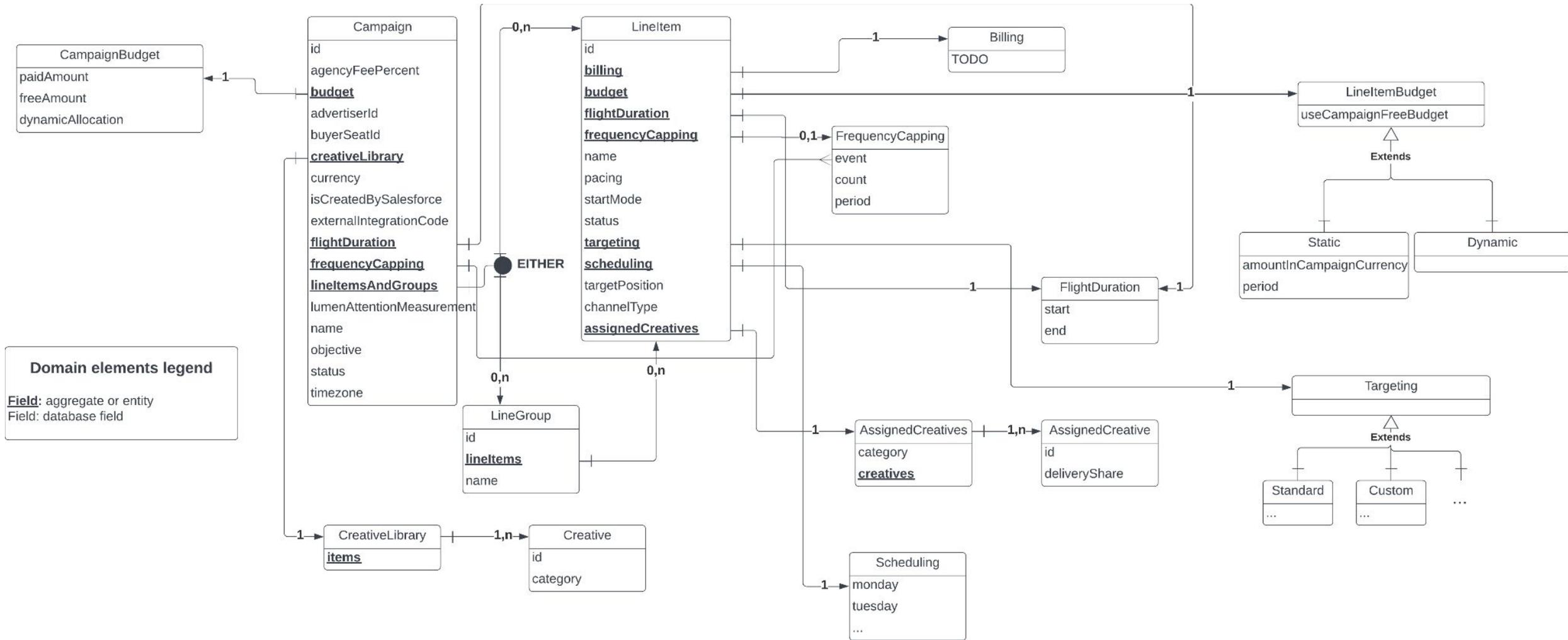
# MERCI

pour votre attention et  
pour vos feedbacks

## Des questions ?

<input type="checkbox"/> 248477	200	xhr	polyfills.3c08d6...	1.4 kB	728 ms	
<input type="checkbox"/> token-info	200	xhr	polyfills.3c08d6...	602 B	60 ms	
<input type="checkbox"/> 12234	200	xhr	polyfills.3c08d6...	513 B	118 ms	
<input type="checkbox"/> 24409	200	xhr	polyfills.3c08d6...	501 B	80 ms	
<input type="checkbox"/> creatives?include=pixelCustoms	201	xhr	polyfills.3c08d6...	1.5 kB	109 ms	
<input type="checkbox"/> creatives?include=pixelCustoms	201	xhr	polyfills.3c08d6...	1.5 kB	106 ms	
<input type="checkbox"/> frontend	200	xhr	polyfills.3c08d6...	450 B	45 ms	
<input type="checkbox"/> videoSettings	201	xhr	polyfills.3c08d6...	1.1 kB	90 ms	
<input type="checkbox"/> videoSettings	201	xhr	polyfills.3c08d6...	1.1 kB	95 ms	
<input type="checkbox"/> 673570	200	xhr	polyfills.3c08d6...	1.5 kB	98 ms	
<input type="checkbox"/> 673571	200	xhr	polyfills.3c08d6...	1.5 kB	100 ms	
<input type="checkbox"/> rules	201	xhr	polyfills.3c08d6...	1.1 kB	78 ms	
<input type="checkbox"/> rules	201	xhr	polyfills.3c08d6...	1.1 kB	80 ms	
<input type="checkbox"/> rules	201	xhr	polyfills.3c08d6...	1.1 kB	77 ms	
<input type="checkbox"/> frontend	200	xhr	polyfills.3c08d6...	450 B	45 ms	
<input type="checkbox"/> insertionOrderLines	201	xhr	polyfills.3c08d6...	1.2 kB	98 ms	
<input type="checkbox"/> insertionOrderLines	201	xhr	polyfills.3c08d6...	1.2 kB	100 ms	
<input type="checkbox"/> insertionOrderLines	201	xhr	polyfills.3c08d6...	1.2 kB	85 ms	
<input type="checkbox"/> plannings	201	xhr	polyfills.3c08d6...	1.1 kB	95 ms	
<input type="checkbox"/> plannings	201	xhr	polyfills.3c08d6...	1.0 kB	90 ms	
<input type="checkbox"/> plannings	201	xhr	polyfills.3c08d6...	1.1 kB	84 ms	
<input type="checkbox"/> broadcasts	201	xhr	polyfills.3c08d6...	1.1 kB	96 ms	
<input type="checkbox"/> broadcasts	201	xhr	polyfills.3c08d6...	1.1 kB	90 ms	
<input type="checkbox"/> broadcasts	201	xhr	polyfills.3c08d6...	1.1 kB	82 ms	
<input type="checkbox"/> volumes	200	xhr	polyfills.3c08d6...	729 B	96 ms	
<input type="checkbox"/> volumes	200	xhr	polyfills.3c08d6...	737 B	91 ms	
<input type="checkbox"/> volumes	200	xhr	polyfills.3c08d6...	734 B	87 ms	
<input type="checkbox"/> broadcastVolumes	201	xhr	polyfills.3c08d6...	1.0 kB	93 ms	
<input type="checkbox"/> broadcastVolumes	201	xhr	polyfills.3c08d6...	1.0 kB	98 ms	
<input type="checkbox"/> broadcastVolumes	201	xhr	polyfills.3c08d6...	1.0 kB	95 ms	
<input type="checkbox"/> cappings	200	xhr	polyfills.3c08d6...	725 B	98 ms	
<input type="checkbox"/> cappings	200	xhr	polyfills.3c08d6...	726 B	96 ms	
<input type="checkbox"/> cappings	200	xhr	polyfills.3c08d6...	723 B	84 ms	
<input type="checkbox"/> diffusionSettings	201	xhr	polyfills.3c08d6...	1.1 kB	114 ms	
<input type="checkbox"/> diffusionSettings	201	xhr	polyfills.3c08d6...	1.1 kB	109 ms	
<input type="checkbox"/> diffusionSettings	201	xhr	polyfills.3c08d6...	1.1 kB	68 ms	
<input type="checkbox"/> insertions?include=diffusionSettings,formatPositions	201	xhr	polyfills.3c08d6...	1.6 kB	130 ms	

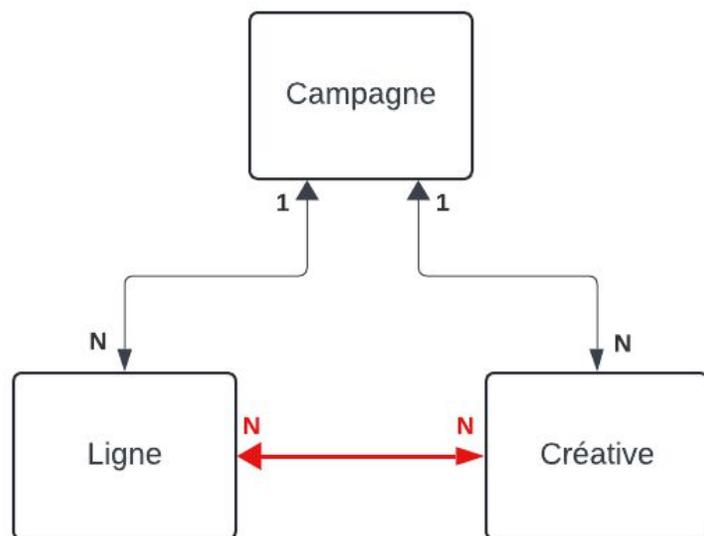
# Faire émerger l'agrégat *Campagne*



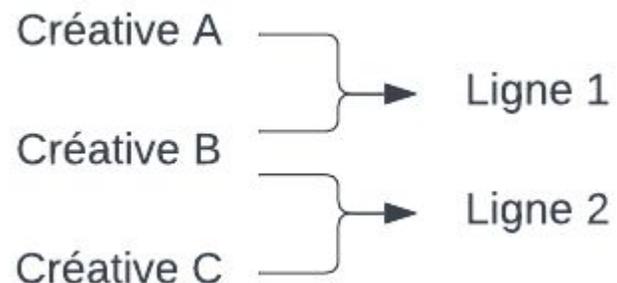
# Déplacer la logique métier vers le backend

Aide à réduire les appels API et donc les performances

Modélisation



Exemple simplifié



4 calls API en //

Chaque call fait une écriture dans la table "Ligne"

Deadlock

## Résolution

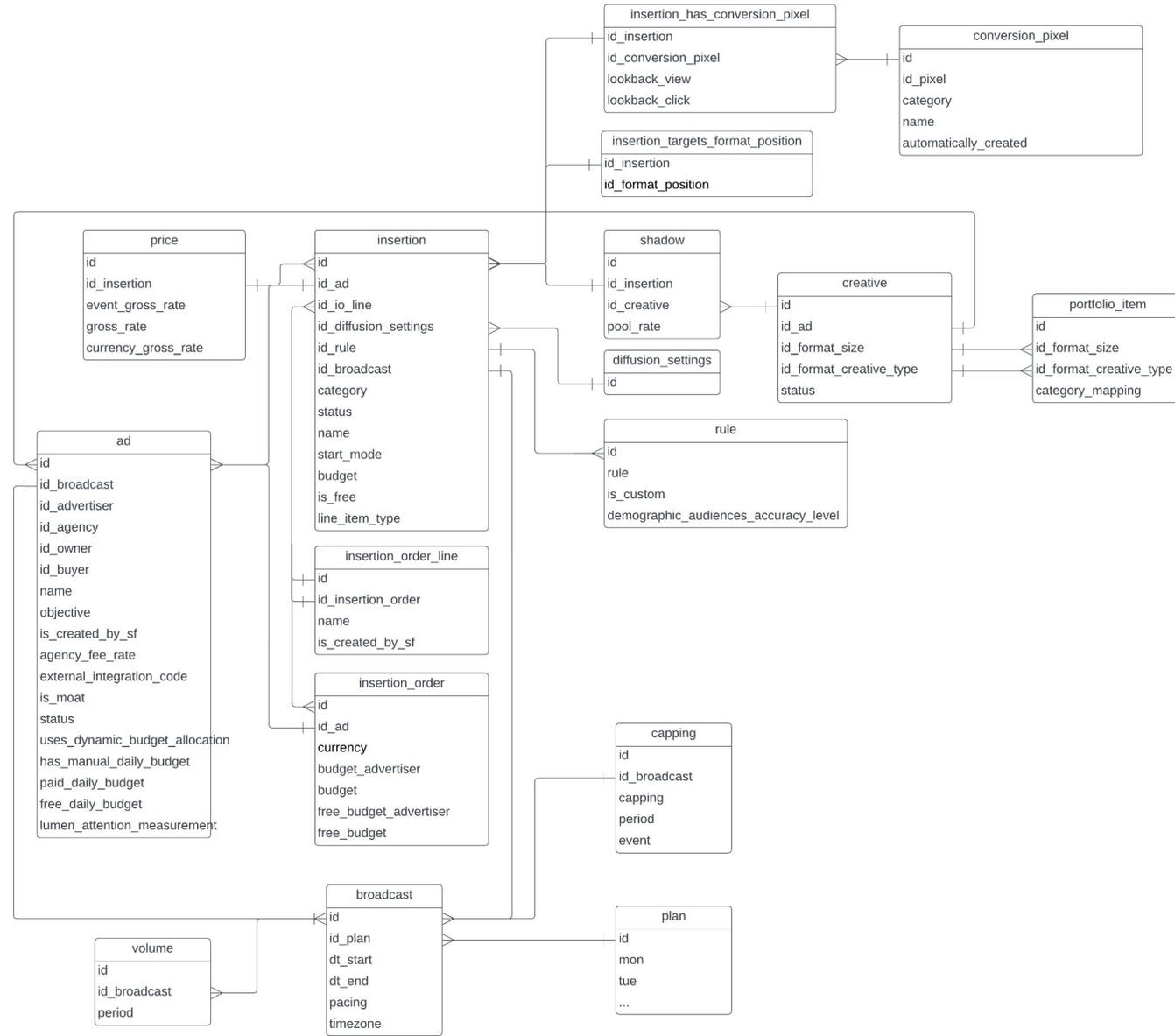
- Séquentiel: Impact les performances et si un call échoue ?
- Batch

Commande dédiée dans la nouvelle api. Commande qui sera temporaire.

Tests automatisés (unitaires / intégration)

Release sous Feature Flag

# Pour s'abstraire du schéma de base de données



# Campaign Edition - Monitoring - Last 24 hours

## Client side metrics

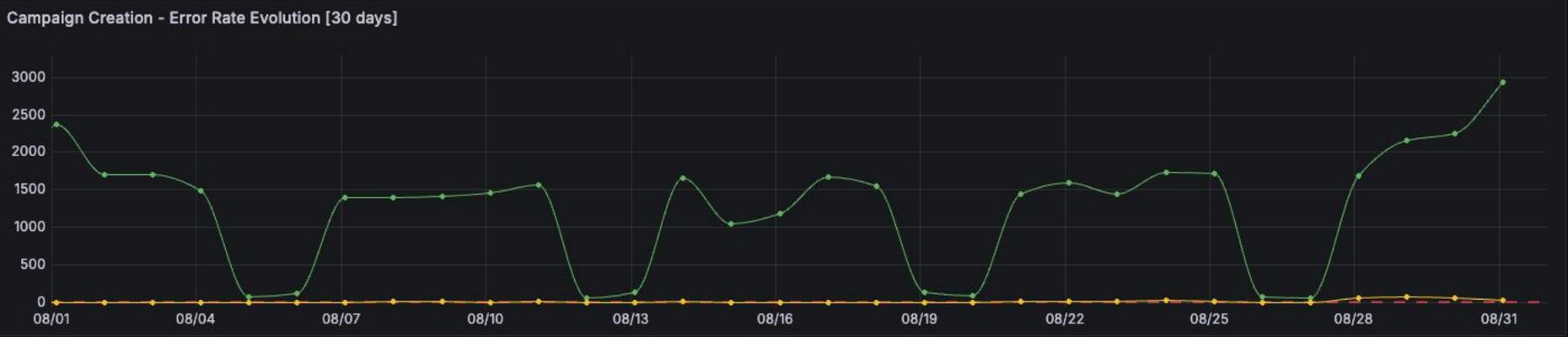
Saving Error Rate

# No data



Saving Error Rate (Legacy)

# 1.03%



Saving Error Rate (Legacy)

# 1.03%



```
message SaveBillingForBrandingCampaignRequest {  
  
    ...  
  
    message BillingForLineItem {  
        required int64 line_item_id = 1;  
  
        oneof model {  
            option (validate.required) = true;  
            CostPerMilleModel cost_per_mille_model = 2;  
            ViewableCostPerMilleModel viewable_cost_per_mille_model = 3;  
            CostPerCompletedViewModel cost_per_completed_view_model = 4;  
            CostPerViewModel cost_per_view_model = 5;  
        }  
    }  
  
    required int64 campaign_id = 2;  
    repeated BillingForLineItem billing_for_line_items = 3 [(validate.rules).repeated.min_items = 1];  
    optional bool dry_run = 4 [default = false];  
}
```



```
message Error {
  oneof error_type {
    CampaignNotFound campaign_not_found = 1;
    CampaignObjectiveMismatch campaign_objective_mismatch = 2;
    CampaignNotSelfServed campaign_not_self_served = 3;
    CampaignNotEditable campaign_not_editable = 4;
    LineItemNotFound line_item_not_found = 5;
    LineItemCampaignMismatch line_item_campaign_mismatch = 6;
    LineItemNotEditable line_item_not_editable = 7;
    LineItemHasUnsupportedCategory line_item_has_unsupported_category = 8;
    CostPerCompletedViewModelIncompatibility cost_per_completed_view_model_incompatibility = 9;
    BillableEventInvalid billable_event_invalid = 10;
    CompleteRateOptimizationIncompatibility complete_rate_optimization_incompatibility = 11;
    BillableEventLockedDuringDelivery billable_event_locked_during_delivery = 12;
    BillingTypeLockedDuringDelivery billing_type_locked_during_delivery = 13;
    CostPerViewModelIncompatibility cost_per_view_model_incompatibility = 14;
  }
}
```