

# ALOM



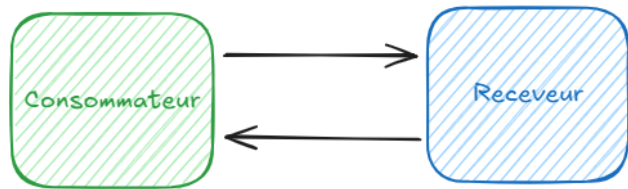
## EDA

*Architectures orientées évènement*

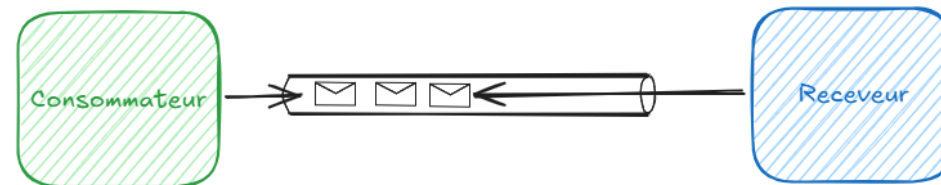
# ARCHITECTURES ORIENTÉES ÉVÈNEMENT

Communications asynchrones via des échanges de message (événements)

🎯 Créer des architectures flexibles, évolutives, réactives



Requête / Réponse



Fire and Forget

# UN ÉVÈNEMENT




Représente un changement d'état, une occurrence dans un système

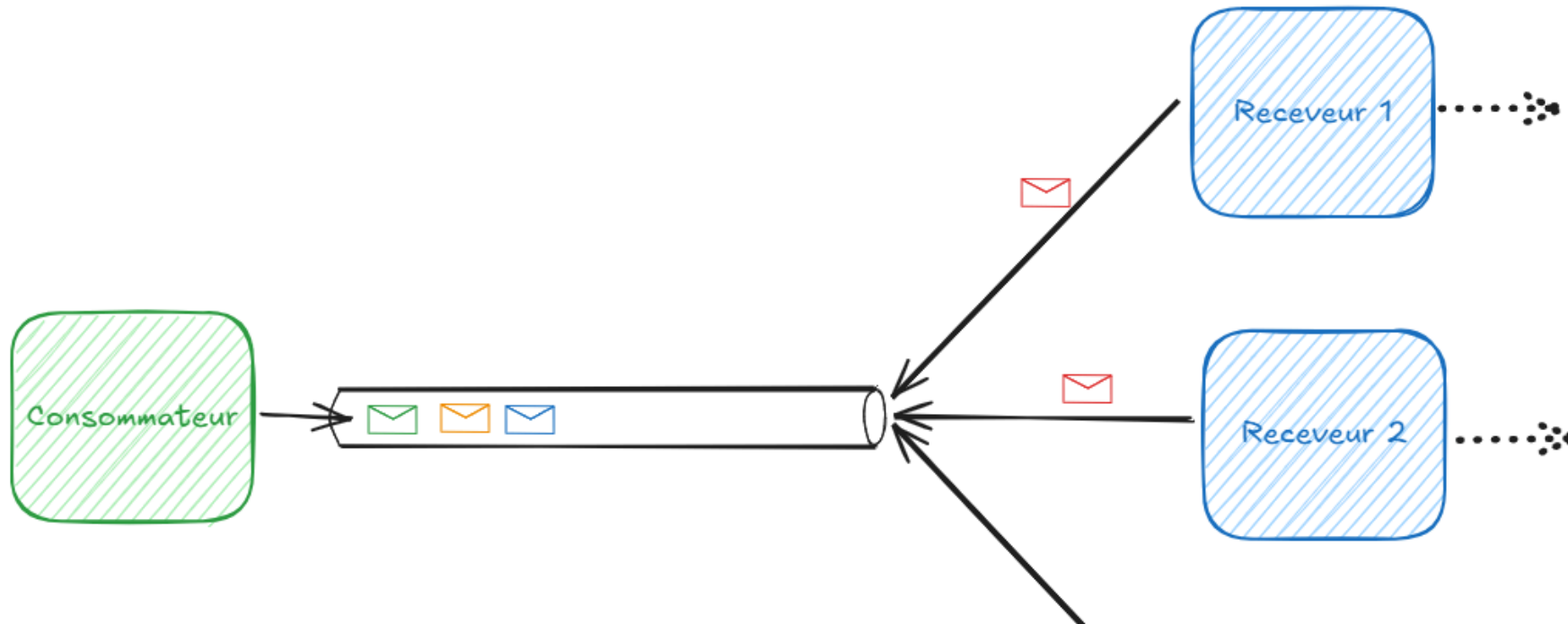
*Commande passée, Paiement effectué, Capteur de passage déclenché, Combat terminé*

Données souvent structurées :

- un en-tête (métadonnées, date d'émission, publisher)
- un payload (contenu de l'évènement)

# ÉLÉMENTS DE L'ARCHITECTURE

-  publisher / producer : émet les évènements
-  subscriber / consumer : reçoit et traite les évènements
-  broker / canaux : transportent les évènements





# BROKERS DE MESSAGE



kafka



RabbitMQ™

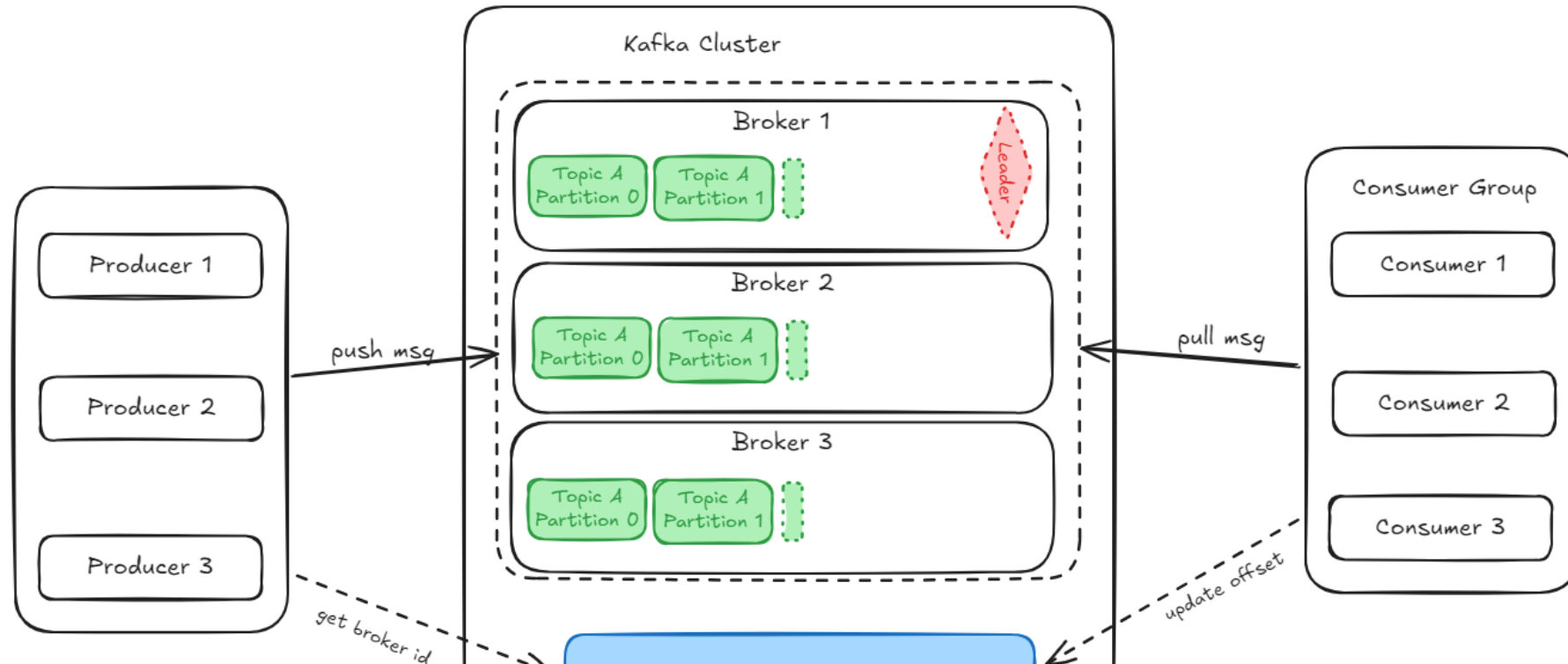


PULSAR



# RÉSILIENCE ET TOLÉRANCE AUX PANNES

-  Persistence des messages
- ↔ Scalabilité horizontale





# AVANTAGES

- découplage
- scalabilité
- flexibilité
- réactivité et temps réel
- résilience






# INCONVÉNIENTS

- complexité de l'orchestration des évènements
- debugging
- cohérence des données



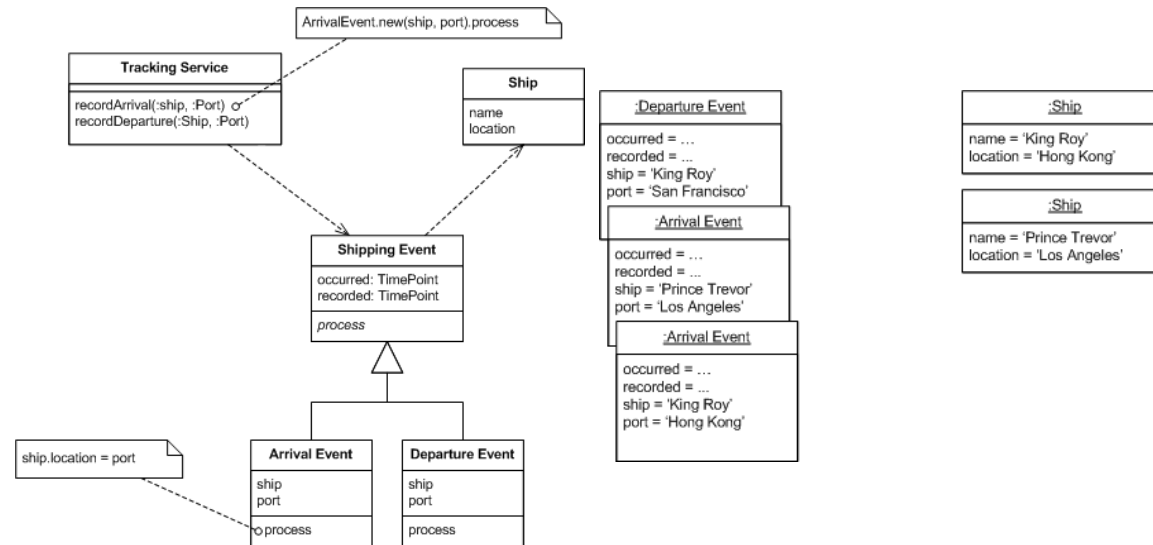


## CAS D'USAGE COURANTS

-  e-commerce : gestion des commandes, mises à jour de stock
-  IoT : traitement de données de capteurs
-  Finance : trading, détection de fraude
-  Micro-services : communication asynchrones
-  CDC : Change-Data-Capture : capture des changement d'une base de données

# PATTERNS

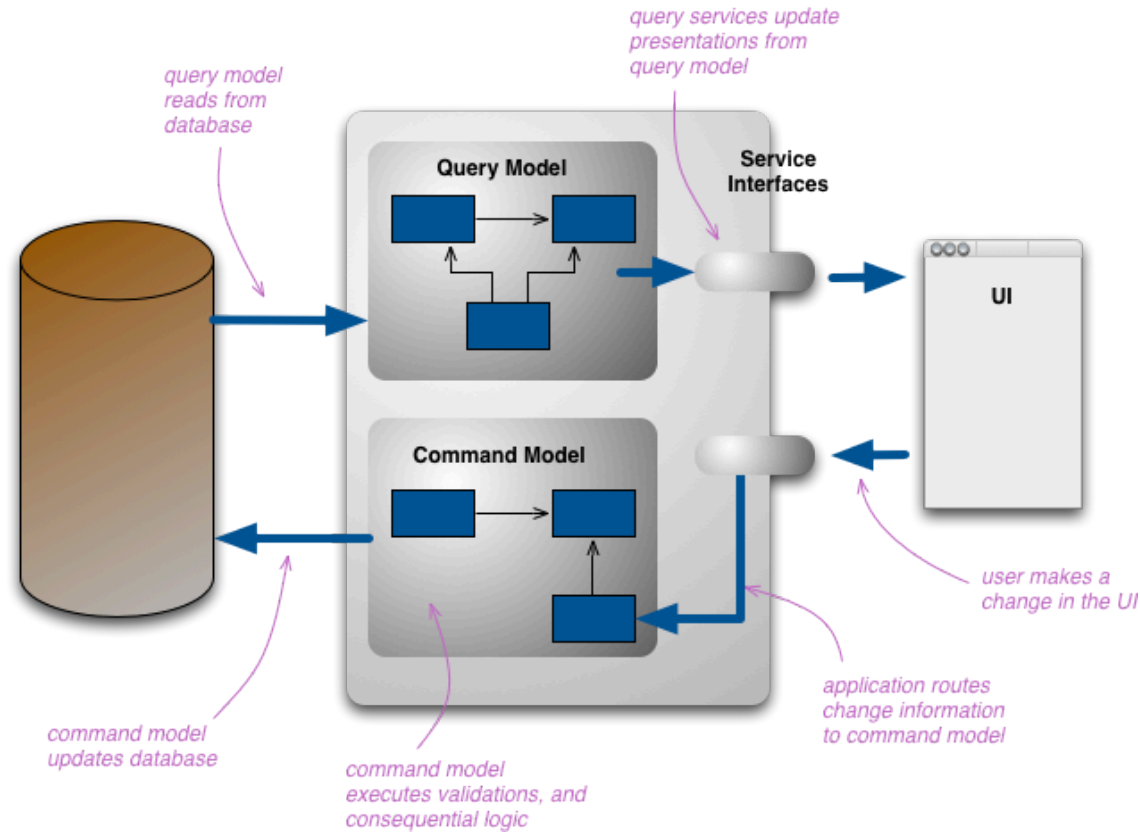
## EVENT SOURCING



- liste des évènements
- possibilité de reconstruire entièrement l'état de l'application
- rejeu d'évènements

# PATTERNS

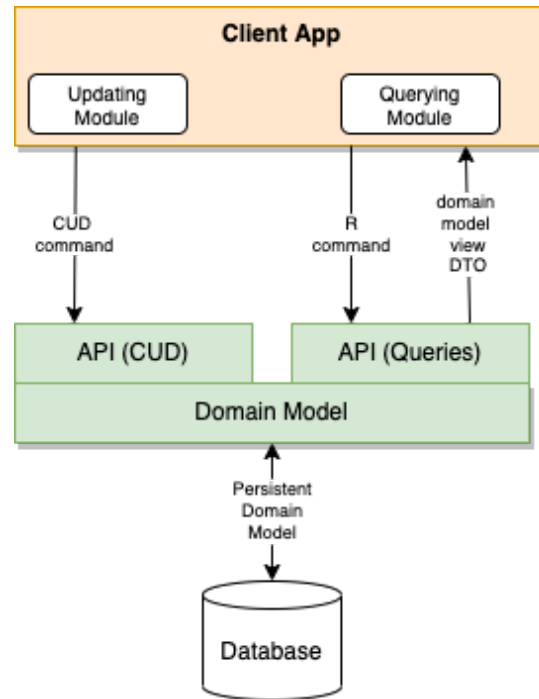
## CQRS



- séparation des *Query* (lecture) et des *Command* (écriture)

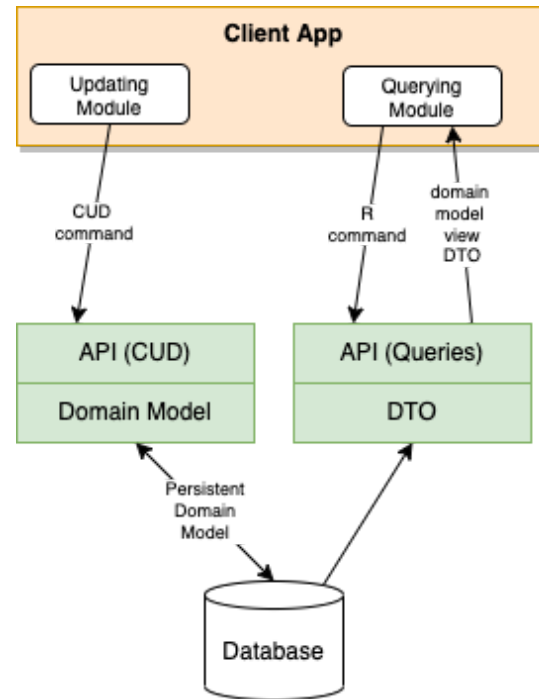


## ROAD TO CQRS (1/3)



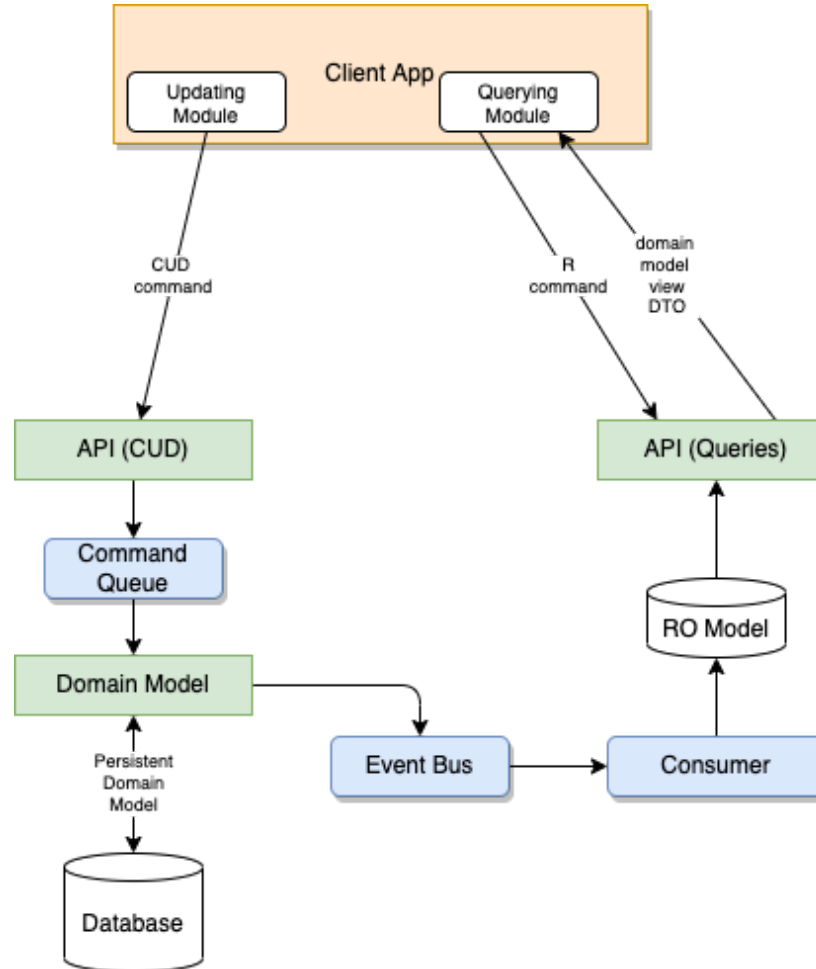


## ROAD TO CQRS (2/3)



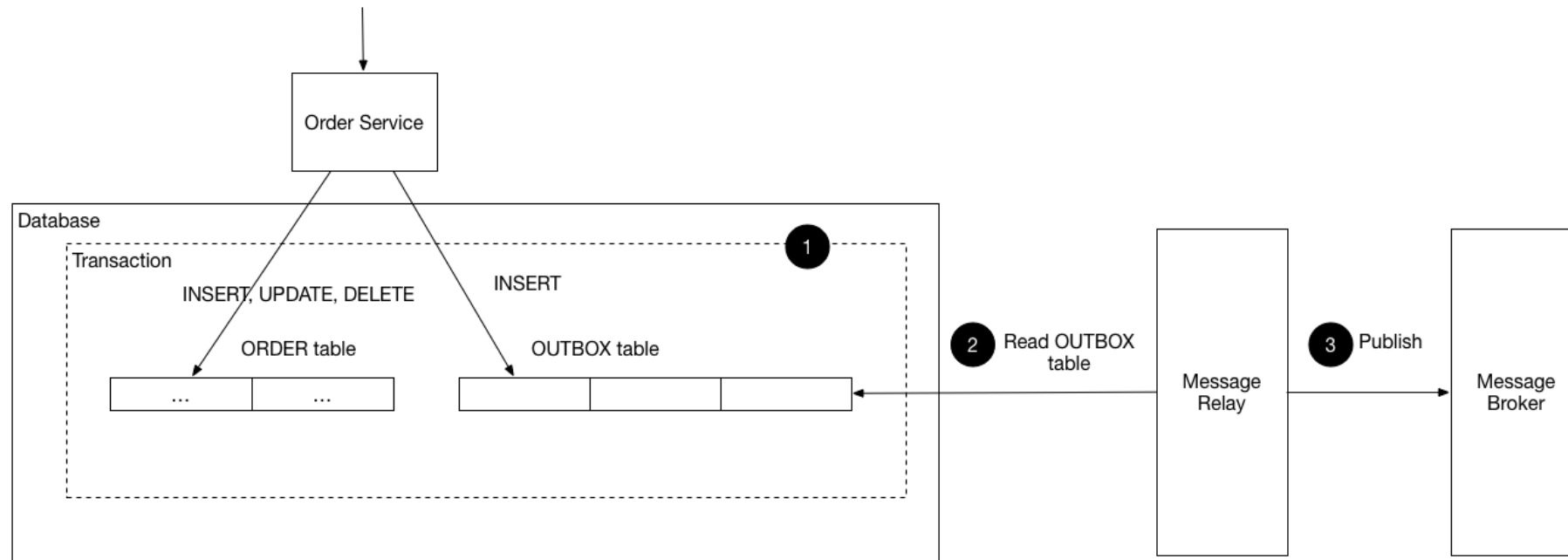


## ROAD TO CQRS (3/3)



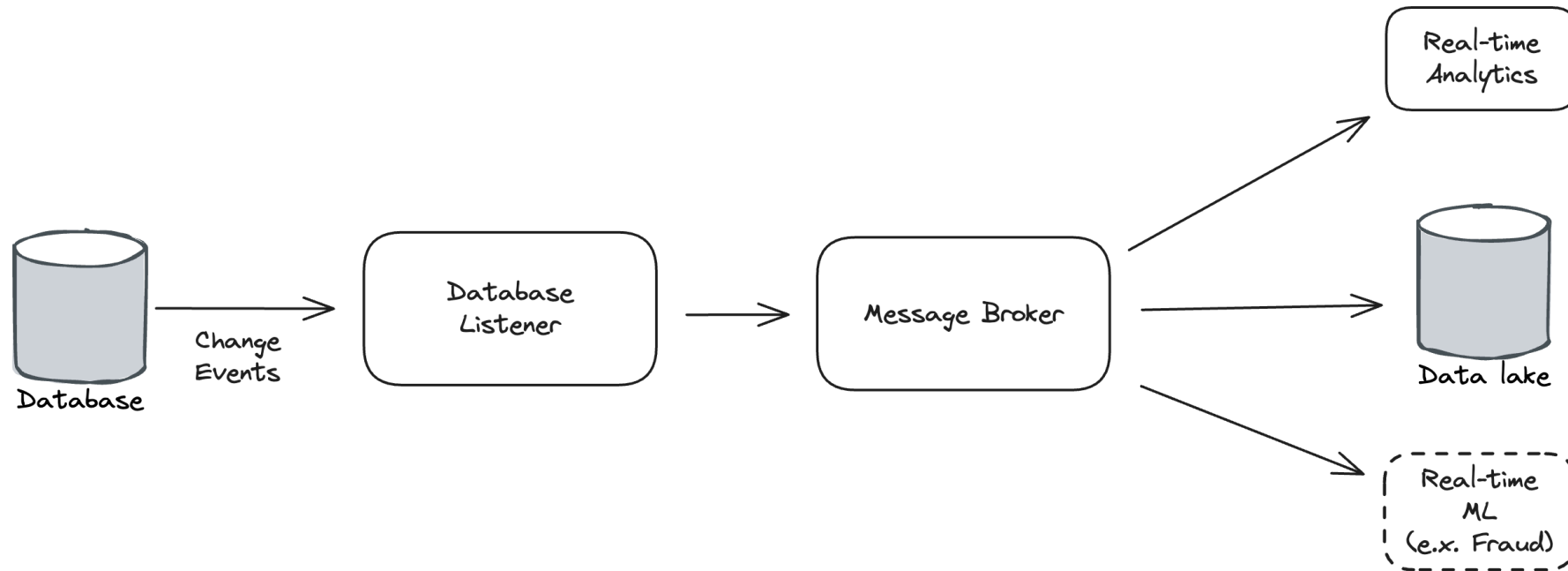
# OUTBOX

rend l'écriture des messages (événements)  
transactionnelle



# CDC : CHANGE-DATA-CAPTURE

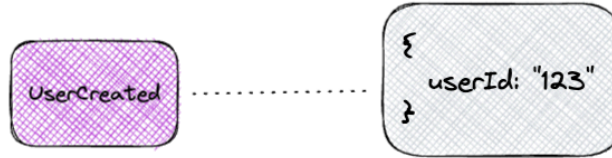
synchroniser des systèmes, intégrer de l'EDA dans du legacy



(Kafka Connect / Debezium)



# EVENT TYPES



## Notification

Notify consumers that something happened  
Normally simple and small payloads.



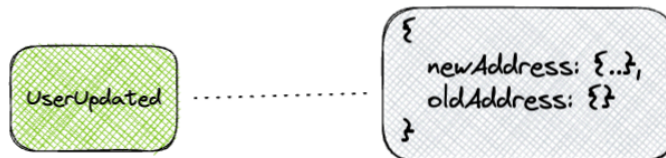
## Event Carried State Transfer

Events that store state / bigger events  
Normally large payload sizes

## Different Event types in Event-driven Architectures

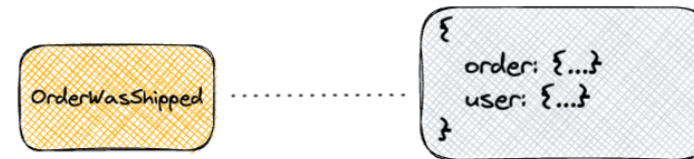
Exploring the different types of events

@boyney123



## Delta Events

Contain the difference between old and new state  
Can be used to share changes



## Domain Events

Focus on domain, things relevant to the business  
Events raised in bounded context

# TP



EDA avec Pulsar