

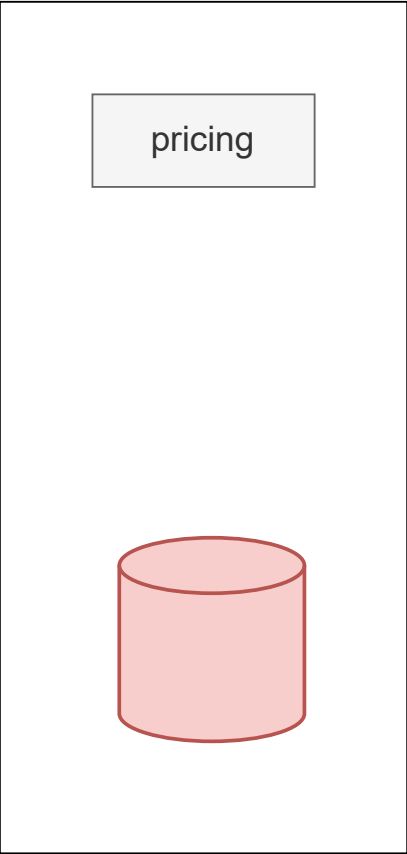
COMMUNICATION

DOMAINE:ECOMMERCE

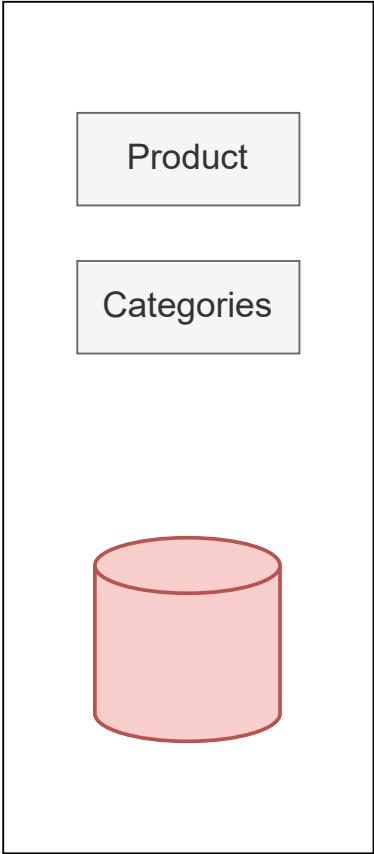
Sous domaines:
User

- . Product
- . Order
- . Inventory
- . Payement
- . Pricing

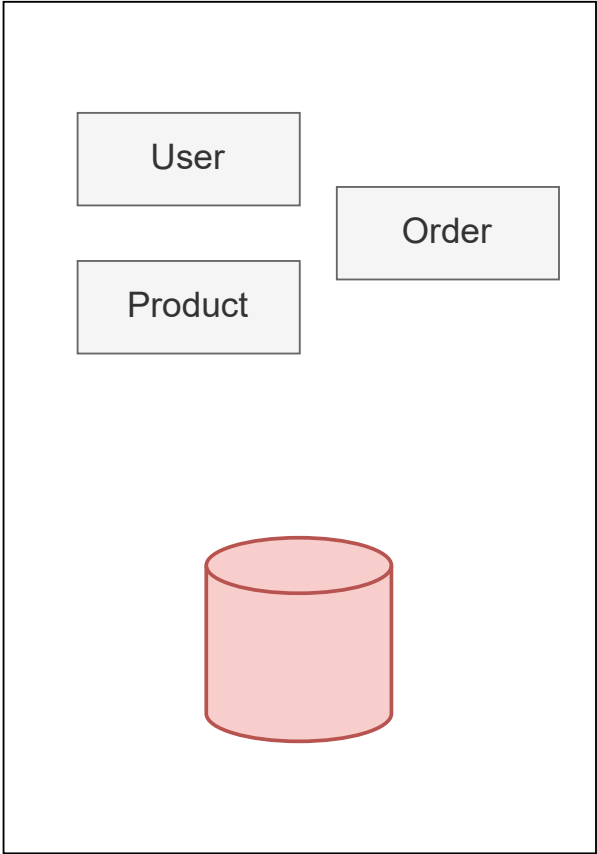
PricingManagement



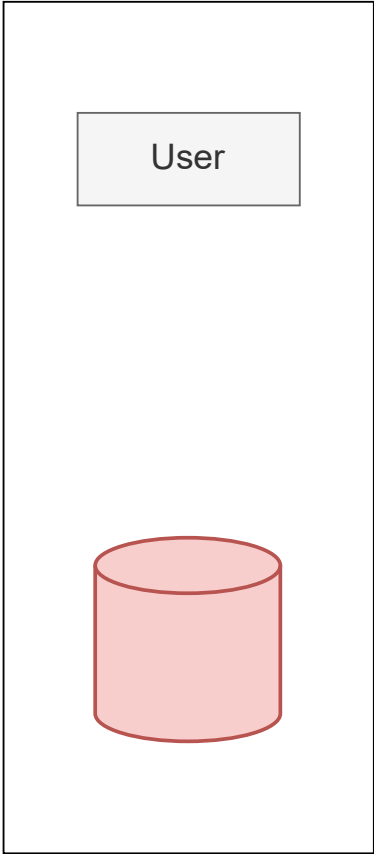
ProductManagement



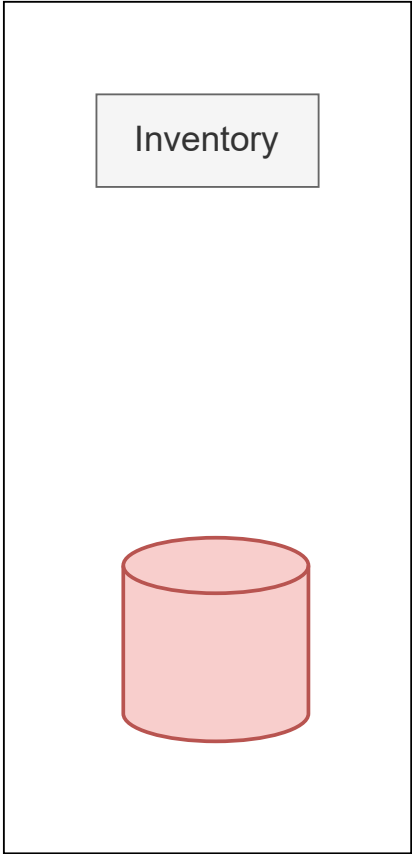
OrderManagemt



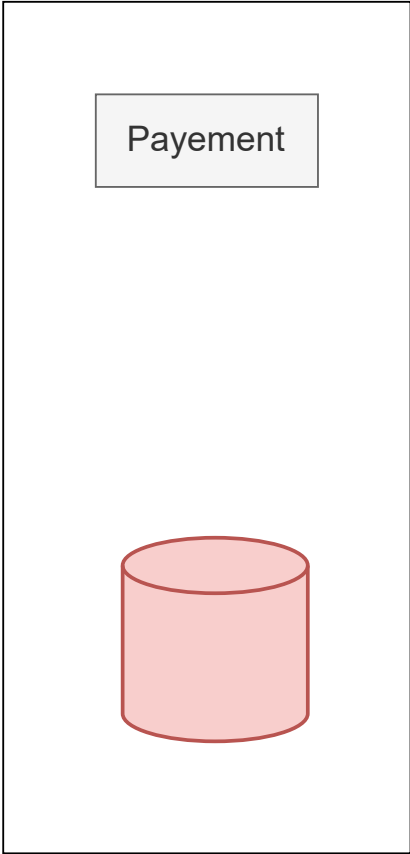
UserManagement



InventoryManagement

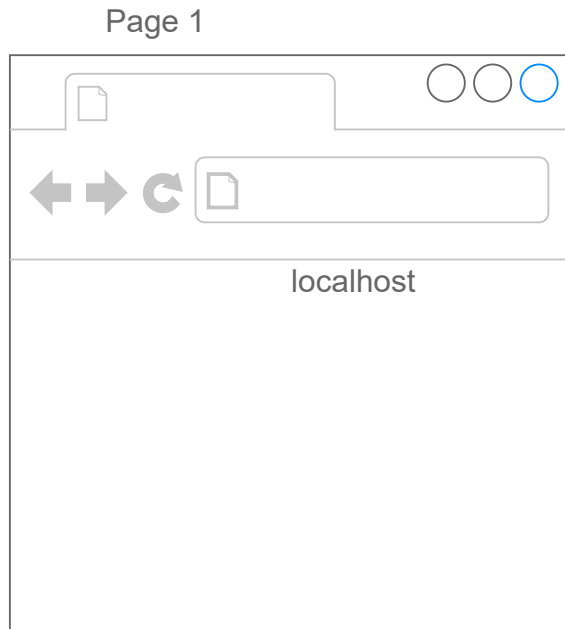


PayementManagement



Synchronous vs. Asynchronous Communication

Sync communication



POST
/order

Order Service

User

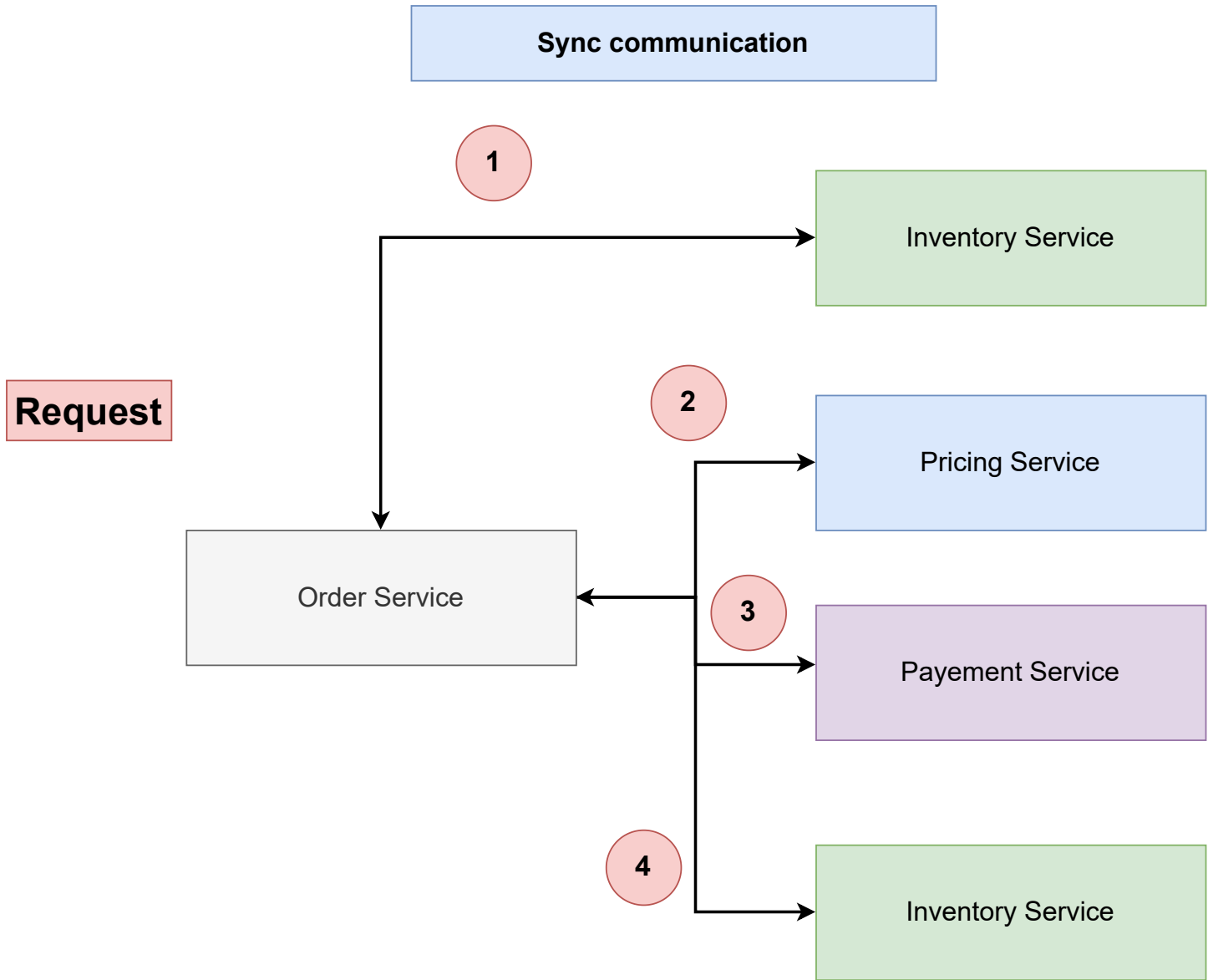
Product

Order

Inventory Service

Inventory

**Hey, Verifie que le produit est
disponible en Stock**



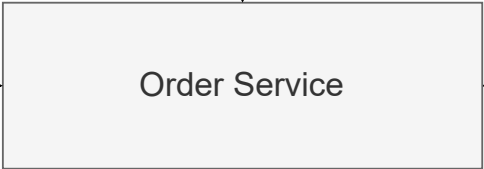
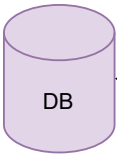
Sync communication



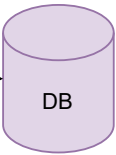
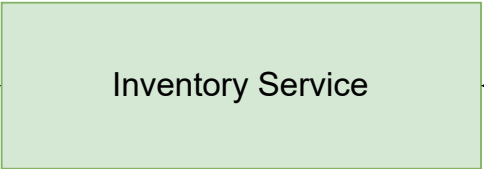
User



1 - L'Utilisateur fait une commande

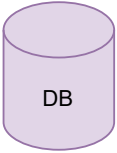
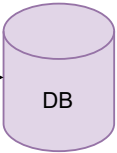
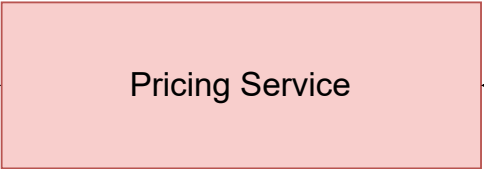


2- Mise a jour d'une stock



Request

3 - Calcul des taxes



Notes on Sync Communication

Conceptuellement facile à comprendre !

Introduit une dépendance entre les services

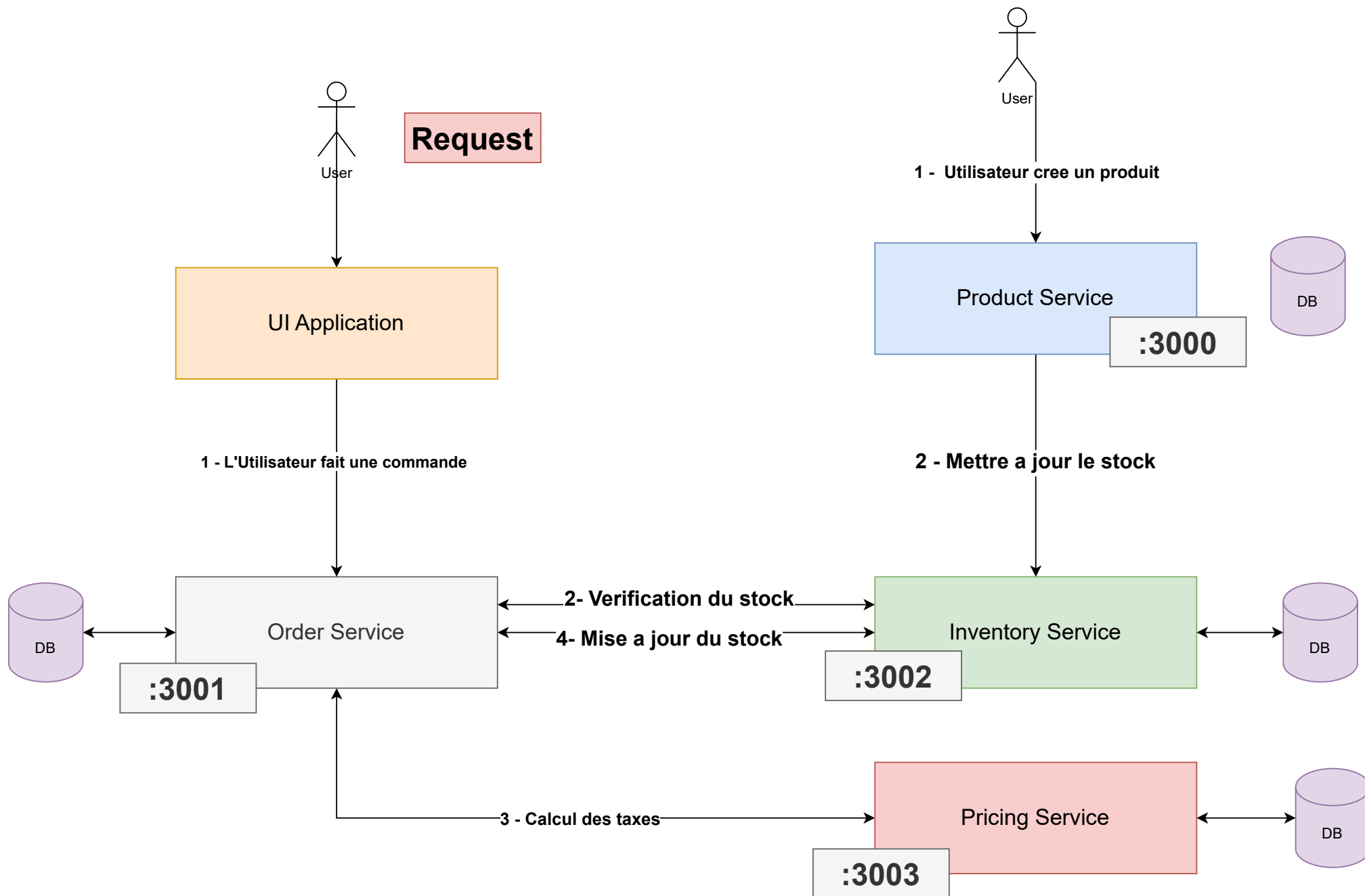
Si une demande interservices échoue, la demande globale échoue.

L'ensemble de la requête n'est pas plus rapide que la requête la plus lente

Peut facilement introduire des réseaux de demandes

DEMO COMMUNICATION SYNCHRONE

DEMO Sync communication



**COMMUNICATION SYNCHORNE
PRODUCT SERVICE ET INVENTORY SERVICE**

DEMO Sync communication

1 - CHARGEMENT DE LA BASE DE DONNEES DES INVENTAIRES

2 - CREATION DES VARAIBLES D'ENVIRONNEMENT

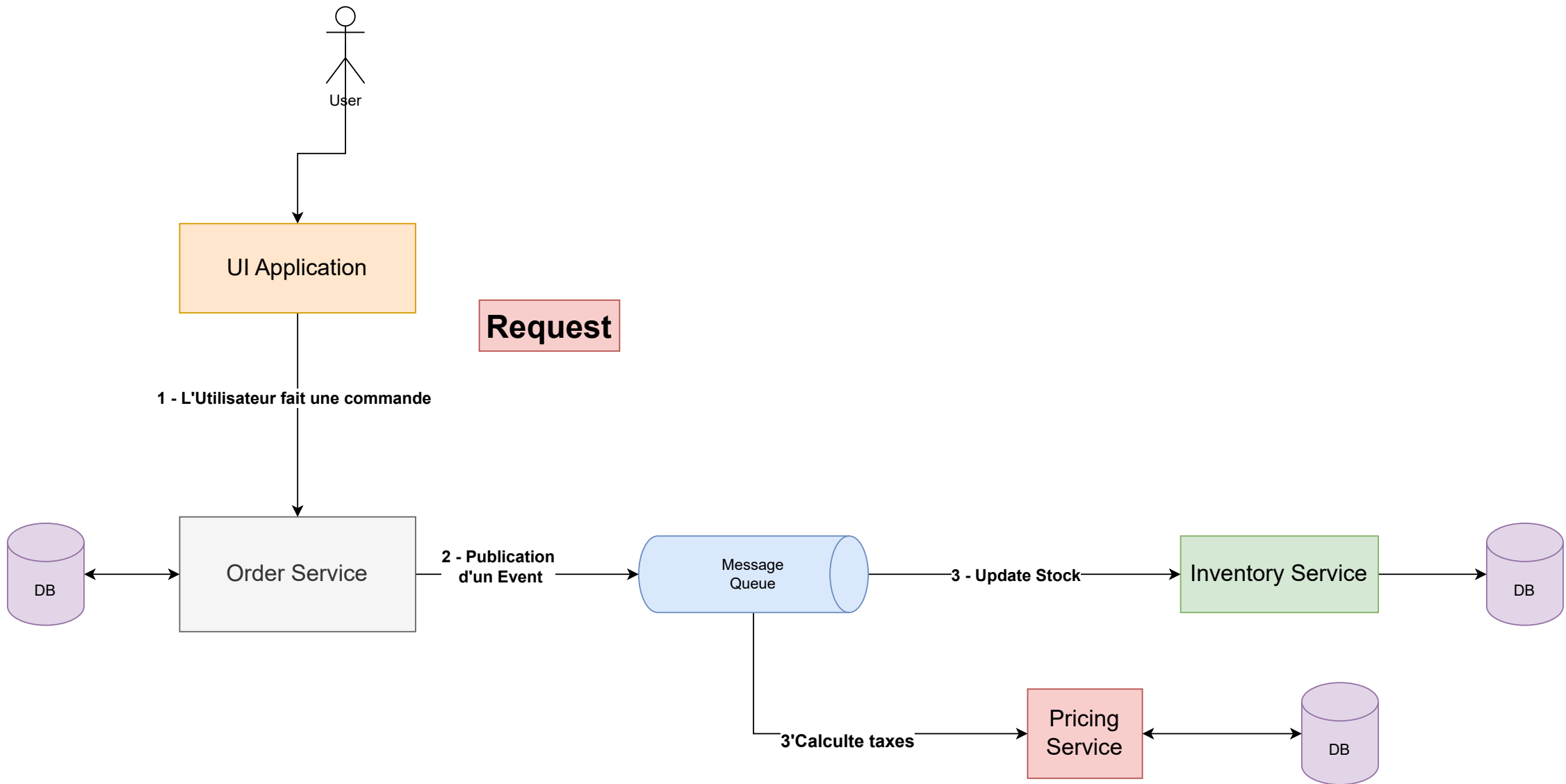
3 - CREATION DU FICHIER CONFIG.JS

4 - ECRITURE DE LA REQUETE

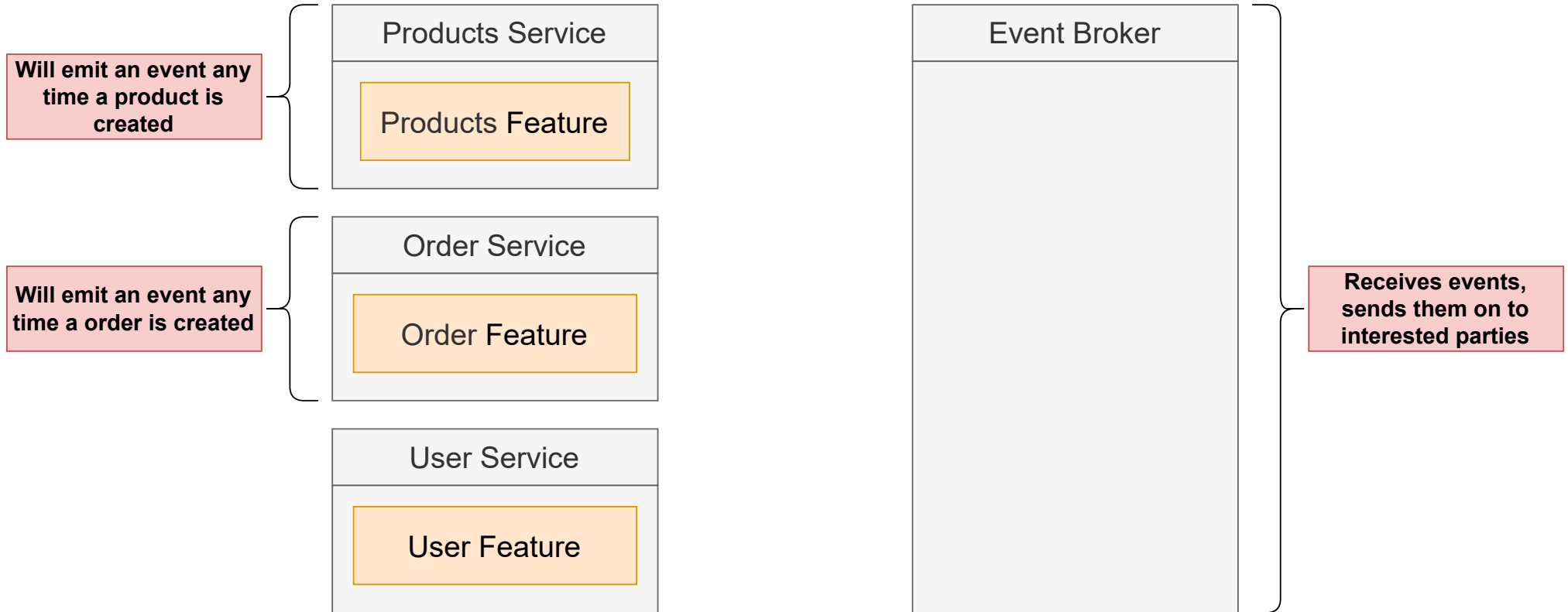
5 - COMMUNICATION AVEC LE PAYEMENT

6 - COMMUNICATION AVEC LE PRICING

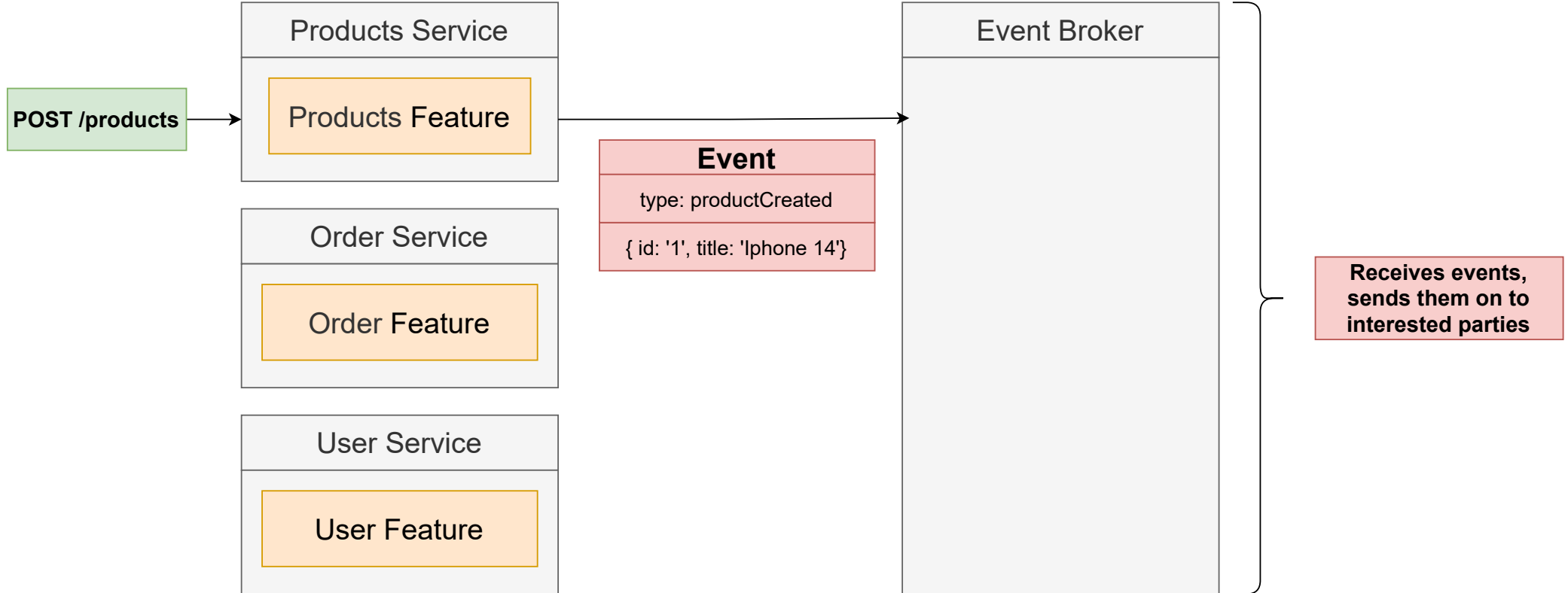
ASYNCRONOUS COMMUNICATION



Async Communication



Async Communication



Notes on Async Communication

Not Wait Response and not have blocked a thread

AMQP (Advanced Message Queuing Protocol)

One-to-one (queue)

One-to-many(topic)

Publish / subscribe

Event Driven Microservices architecture

Notes on Async Communication

Query Service has zero dependencies on other services!

Query Service will be extremely fast!

Data duplication.

Harder to understand

EVENTS

Event: Principales caractéristiques

- Il s'est passé quelque chose
- Asynchrone
- Ne renvoie jamais de réponse
- Le service appelant n'a aucune idée de qui gère l'événement

```
const message = JSON.stringify({  
  action: "inventoryUpdated",  
  productId: productId,  
  payload: {  
    quantity: -quantity,  
  }  
});
```

Events

UserCreated

UserUpdated

OrderCreated

OrderCancelled

OrderExpired

ProductCreated

ProductUpdated

PaymentCreated

Channel / Message Broker

Types of Message Broker

AMQP (Exemple RABBITQM)

QUEUES

Exchanges

Bindings

Durability and Acknowledgements

LOG-BASED (Exemple Kafka)

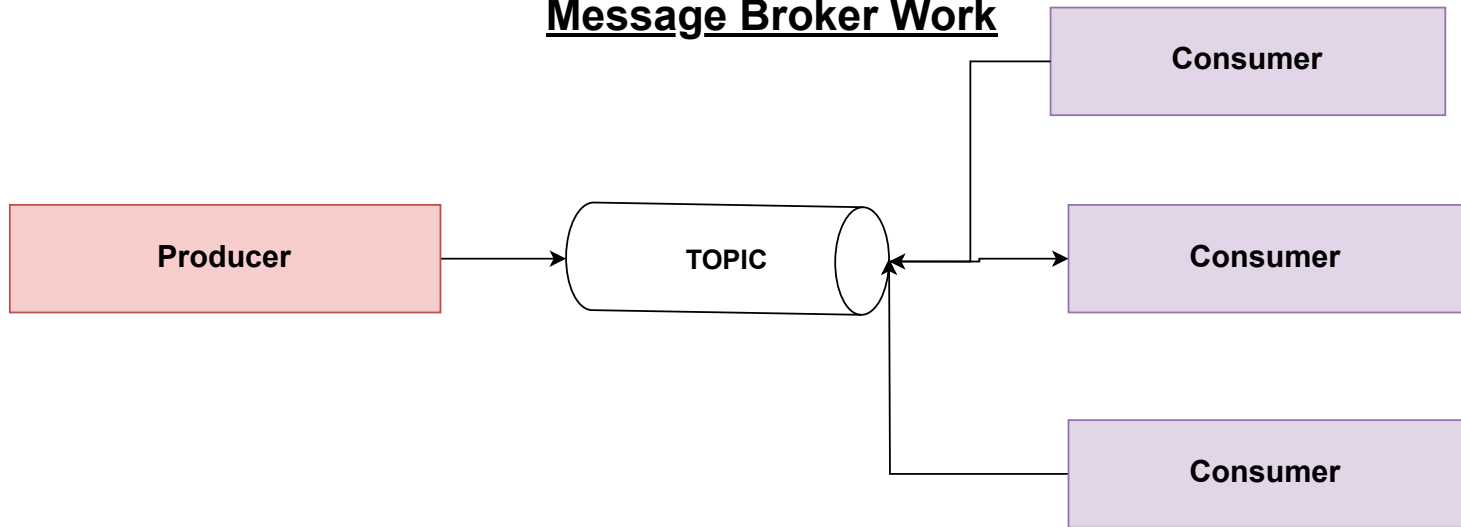
Topics

Partitions

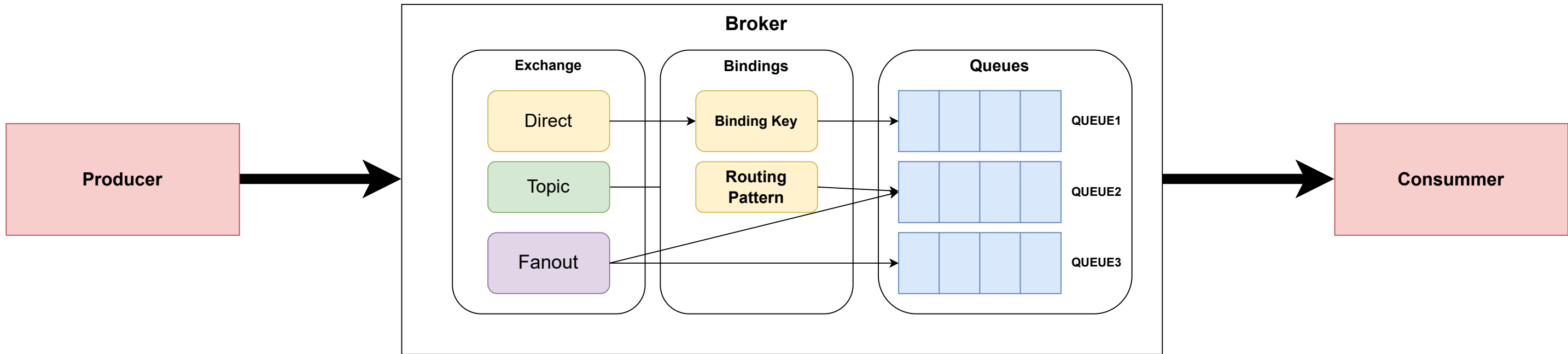
Offset Management

Replication

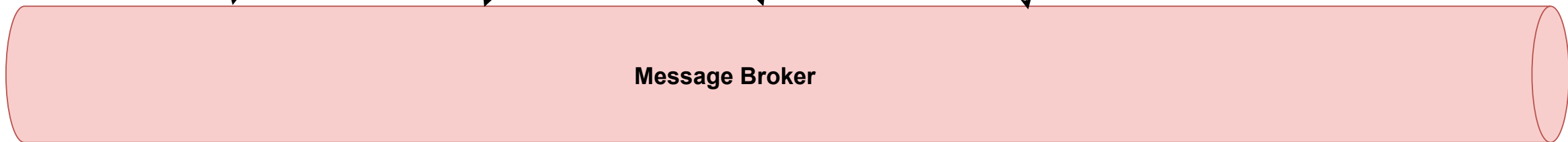
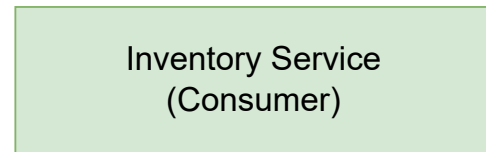
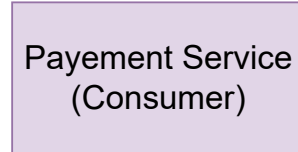
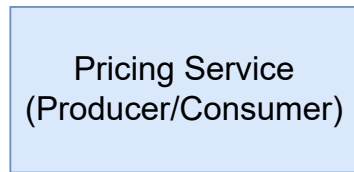
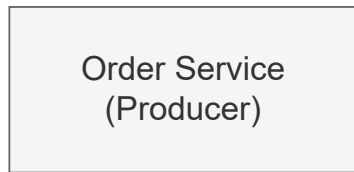
Message Broker Work



RabbitMQ Message Broker



EDA



DEMO COMMUNICATION ASYNCHORNE