

Architecture Microservice Principes et Partiques avec NodeJS

*Regis Atemengue
@regis_ate
www.regisatemengue.com*

Contributeurs

DAVE CHEDJOUN

ABDEL RAZAK

FOLONG EMERSON

CEDRIC MESSI SYLVESTRE



Diagram

Ressources pour suivre le workshop

https://regisatemengue.com/meetup/Architectures_microservices_principes_et_partiques/

Diagrammes

Diagrammes Jour 1

Diagrammes Jour 2

Code de Base / S

<https://regisatemengue.com/codebase/architec>

Télécharger
le code base sur github

api-fonda-start

docker

local-start

kubernetes

sync-com-start

logging

async-com-start

monitoring

api-gateway-
start

testing

auth-start

deploy

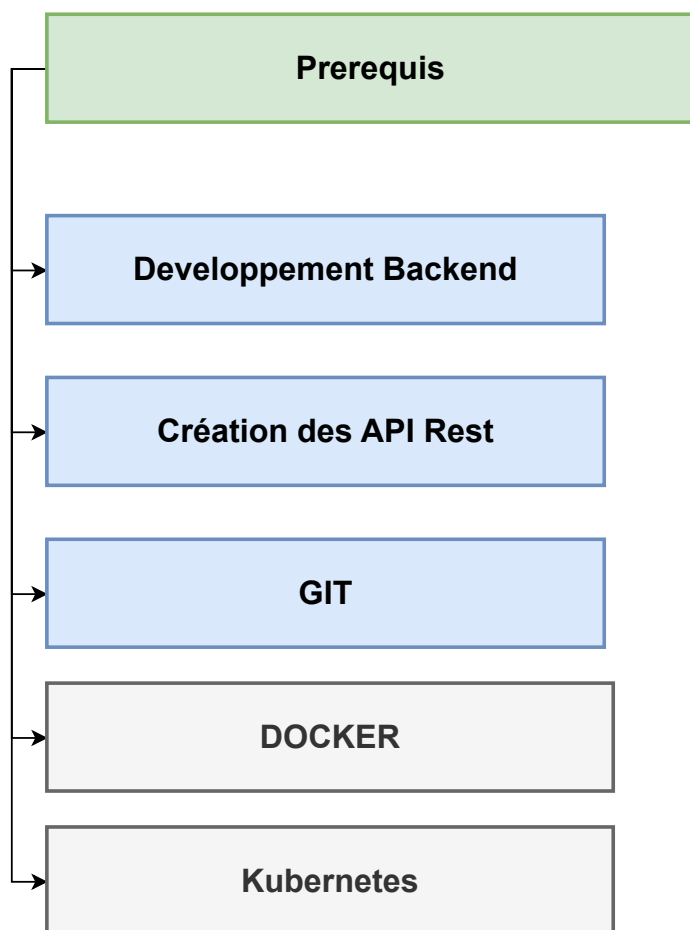


nippet du projet
structures_microservices_principes_et_partiques/



**Accéder aux
Snippets**





Environnement de Travail

node v20.11.1

Laravel 10.10

Java 14 ou Supérieur

Editeur de Code

Visual Studio Code

Autres

Base de données

mongo DB
(local/ image docker)

Postgresql
(local/ image docker)

Interface Graphique pour le BD

PgAdmin

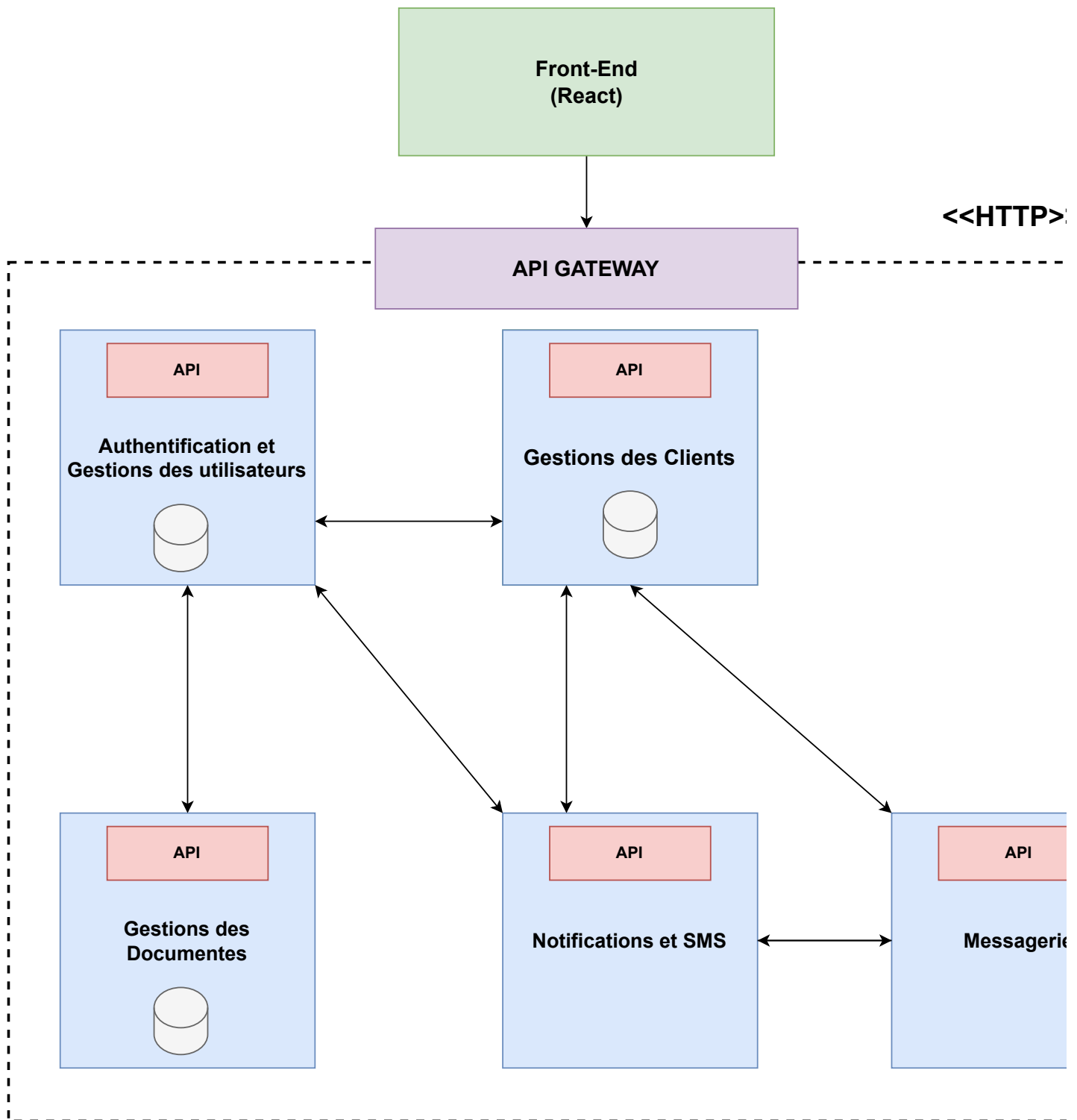
mongodb compass / Studio 3T

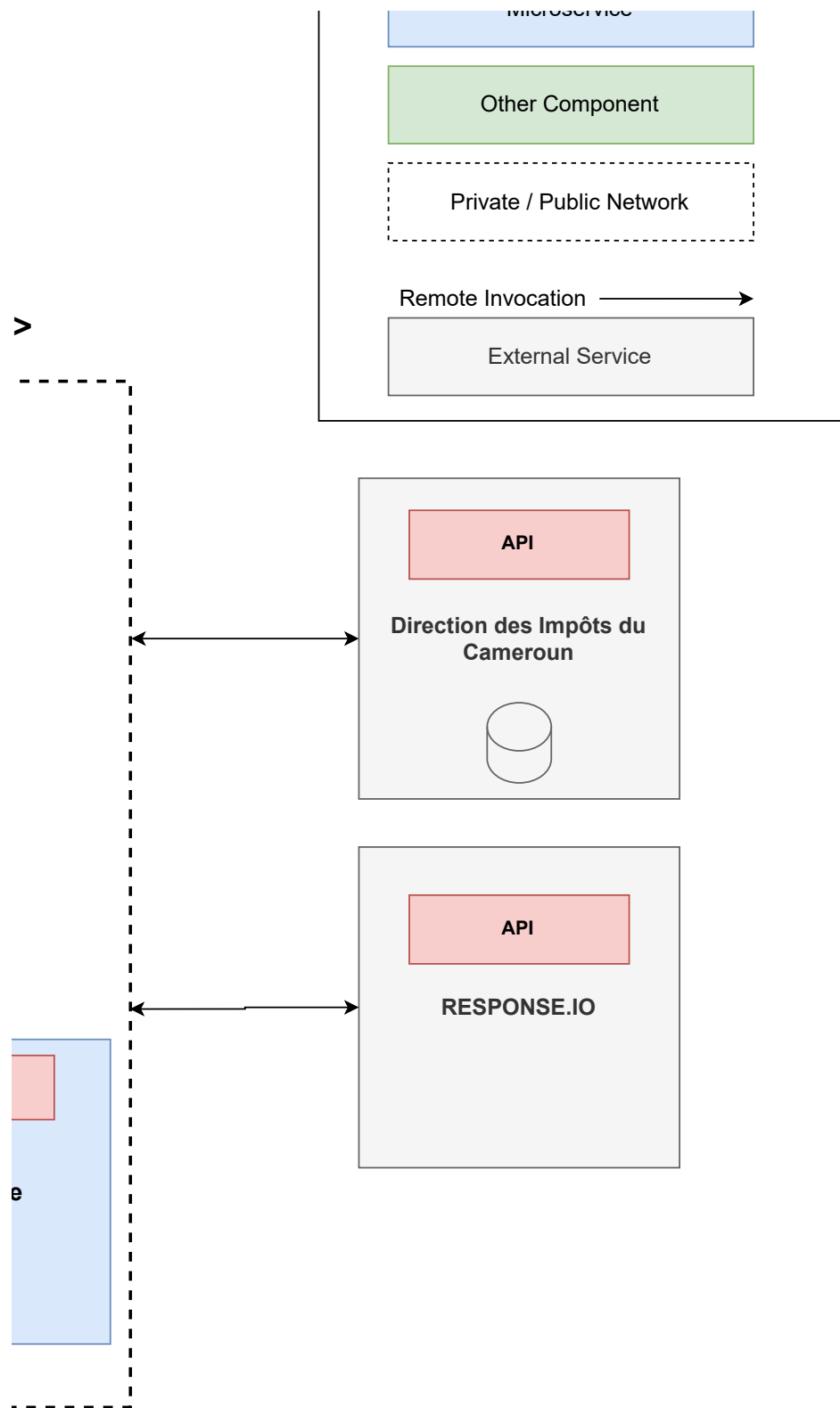
Extension VS Code

- Extension VS Code: Rest CLIENT

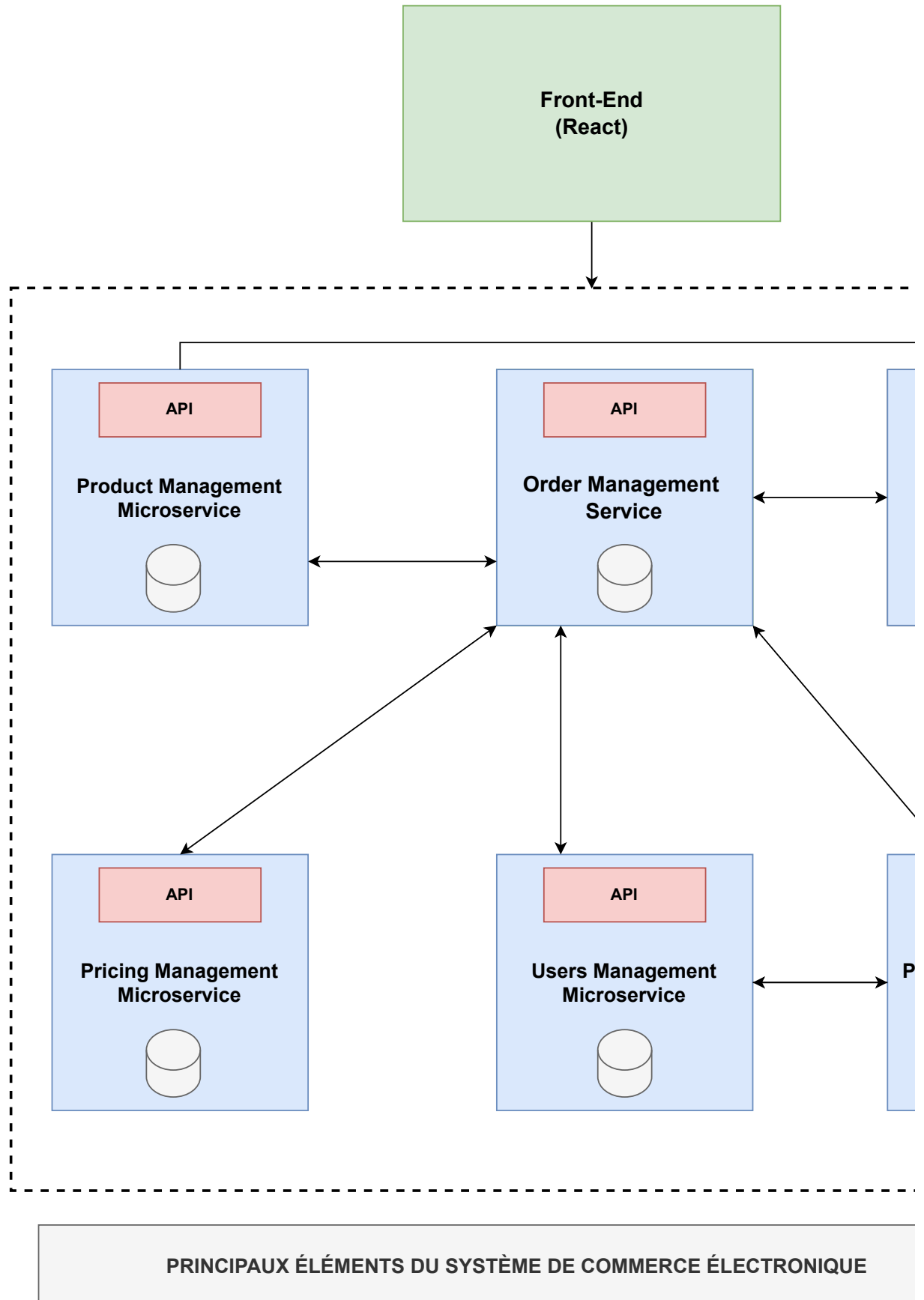
Legend

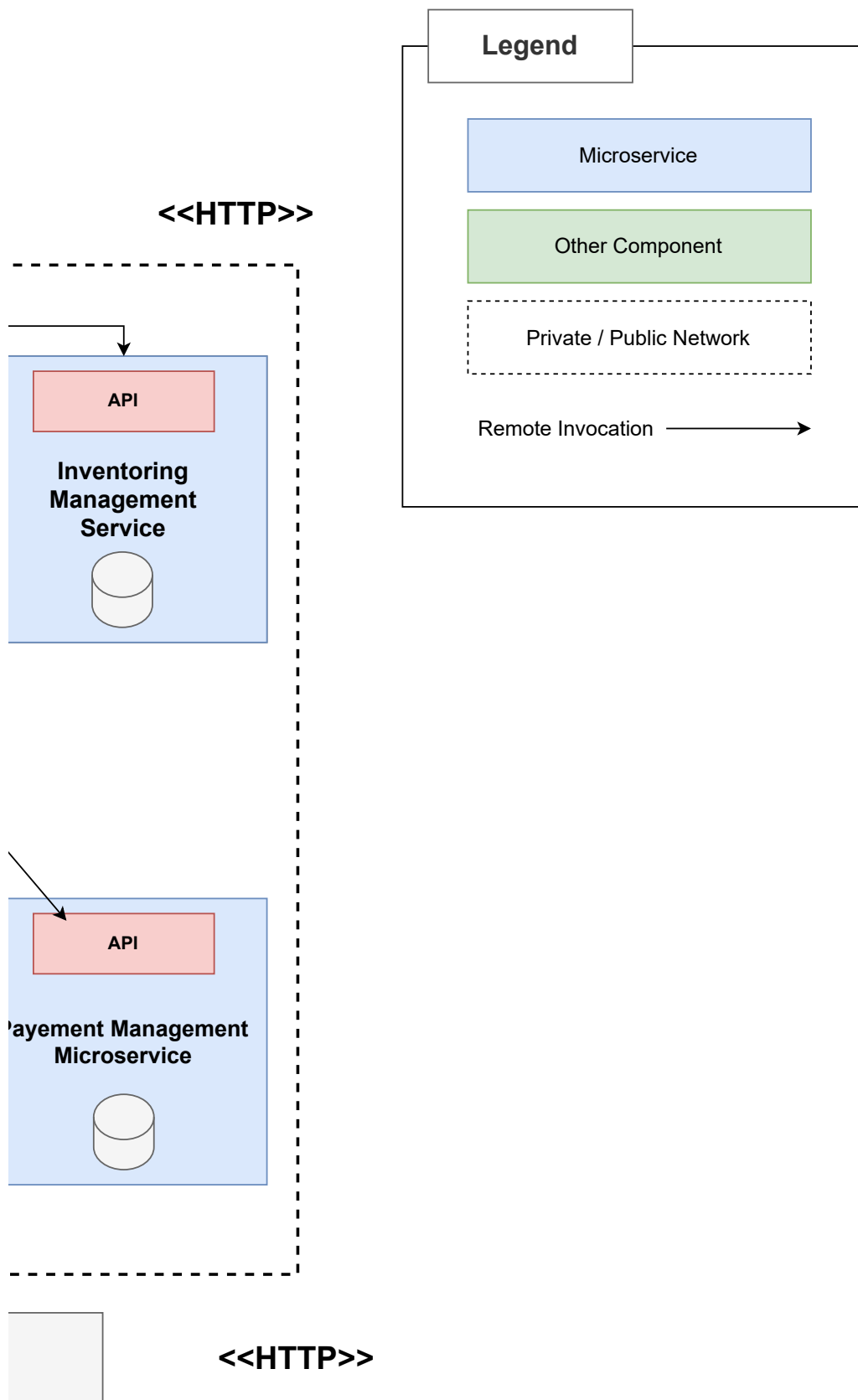
Microservice





Architecture de départ





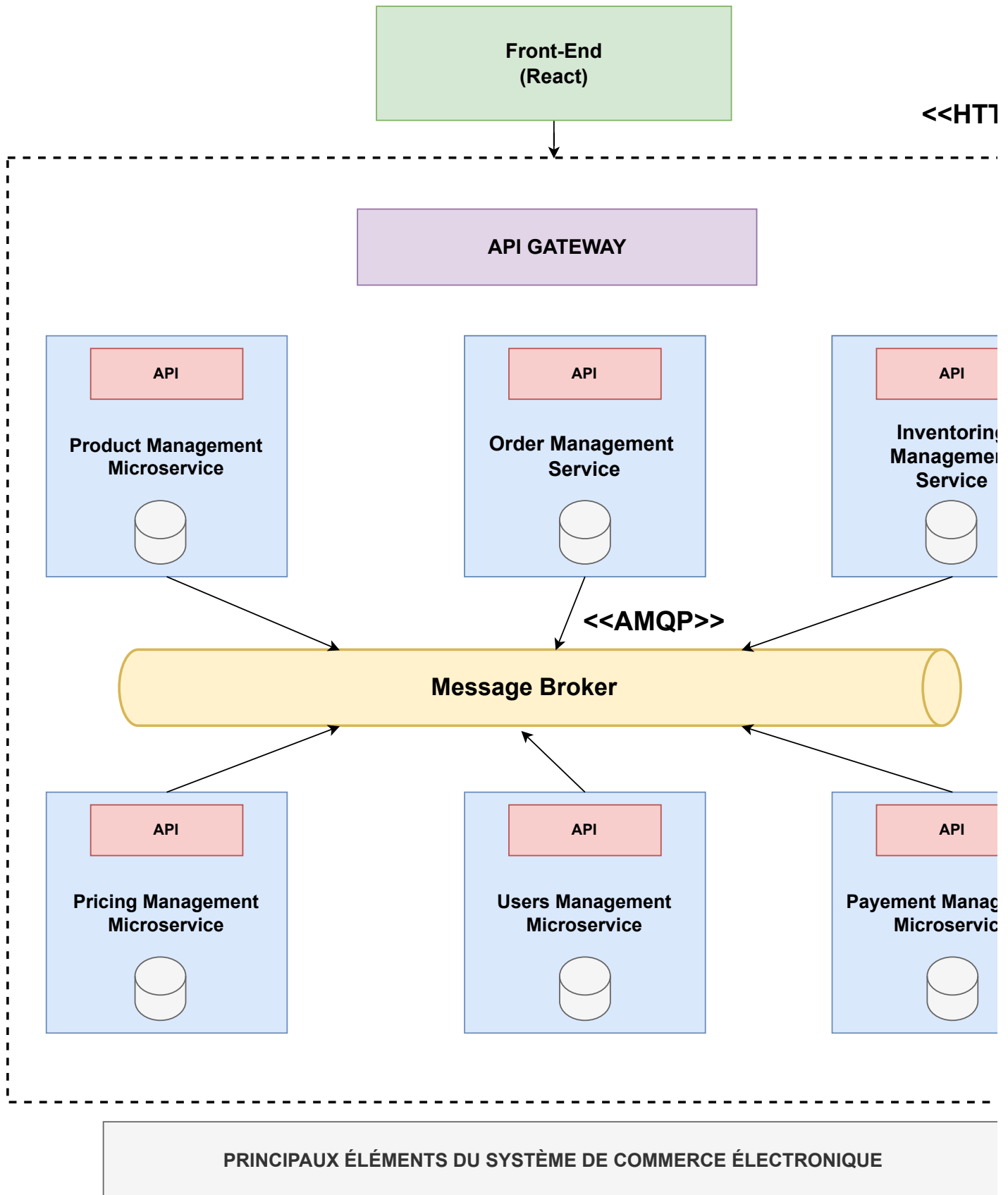
Profils

**Développeurs
débutants/intermédiaires**

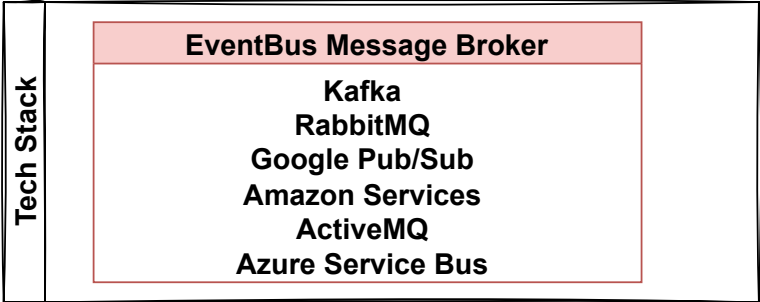
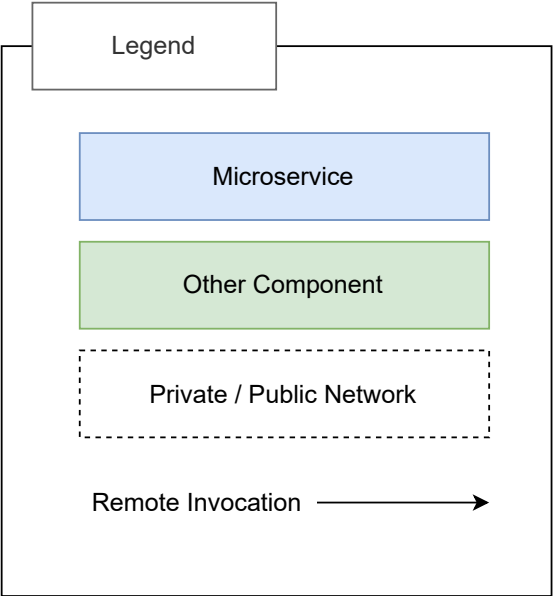
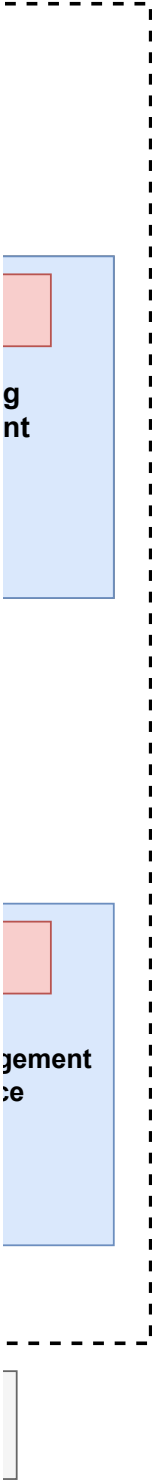
Développeurs expérimentés

**Etudiants en Informatique L2 L3
M1 et M2**

Architecture Finale
Communication Asynchorne



[PS>>



ARCHITECTURE

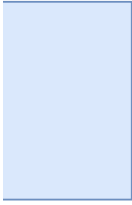
```
graph TD; A[ARCHITECTURE] --> B[Vision]; A --> C[Stru]; B --> D[Le processus de création d'une architecture consiste à traduire les pilotes architecturaux (exigences fonctionnelles, attributs de qualité, contraintes et principes) en une solution technique, créant ainsi une feuille de route ou une vision technique.]; E[Il s'agit de la décoration d'un ensemble d'éléments et de leurs interactions/relations];
```

Vision

Le **processus de création** d'une architecture consiste à traduire les pilotes architecturaux (exigences fonctionnelles, attributs de qualité, contraintes et principes) en une solution technique, créant ainsi une feuille de route ou une vision technique.

Stru

Il s'agit de la **décor** d'un ensemble d'éléments et de leurs **interactions/relations**.



Structure

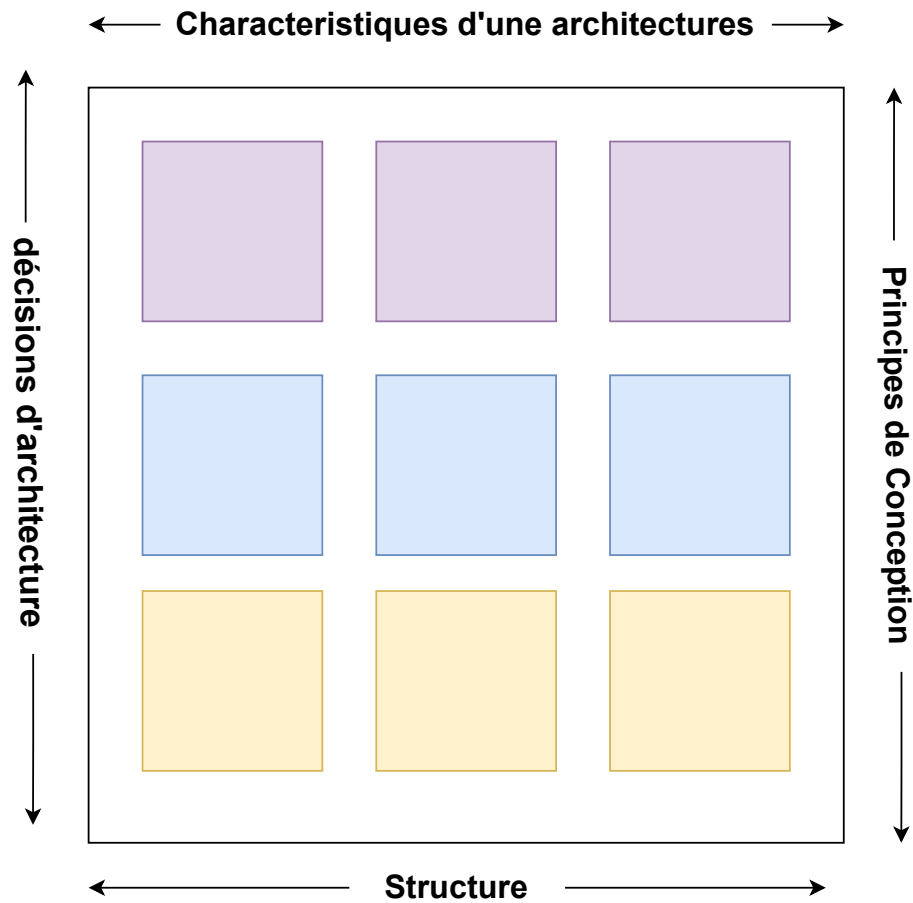


Décomposition d'un produit en
éléments plus petits et des
Relations entre ces éléments.

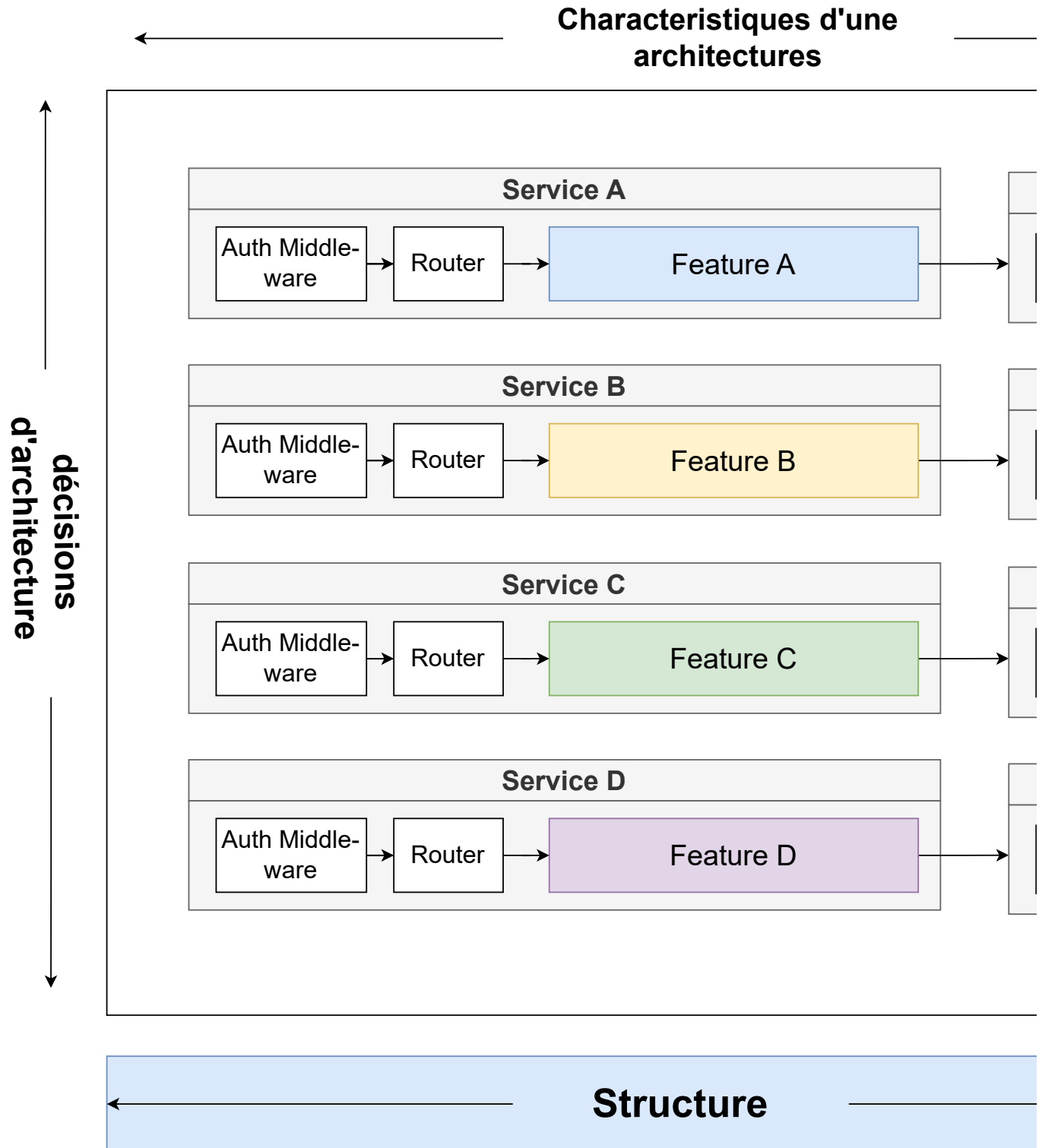
Architecture Logiciel

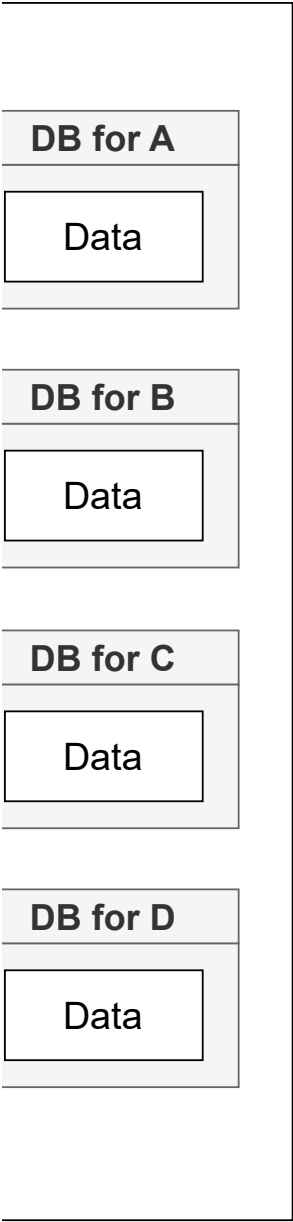
L'architecture logicielle se compose de la **structure du système** combinée aux **caractéristiques architecturales** ("qualités/capacités") que le système doit prendre en charge, aux **décisions d'architecture** et, enfin, aux **principes de conception**.

Architecture Logiciel



Architecture Logiciel





Principes de
Conception



Styles Architecturaux/ Structure

Monolithique

Layered Architecture

**SOA (Service Oriented
Architecture)**

Microservices

Event-Driven Architecture

Monolithe/ Monolithique

Se

Request

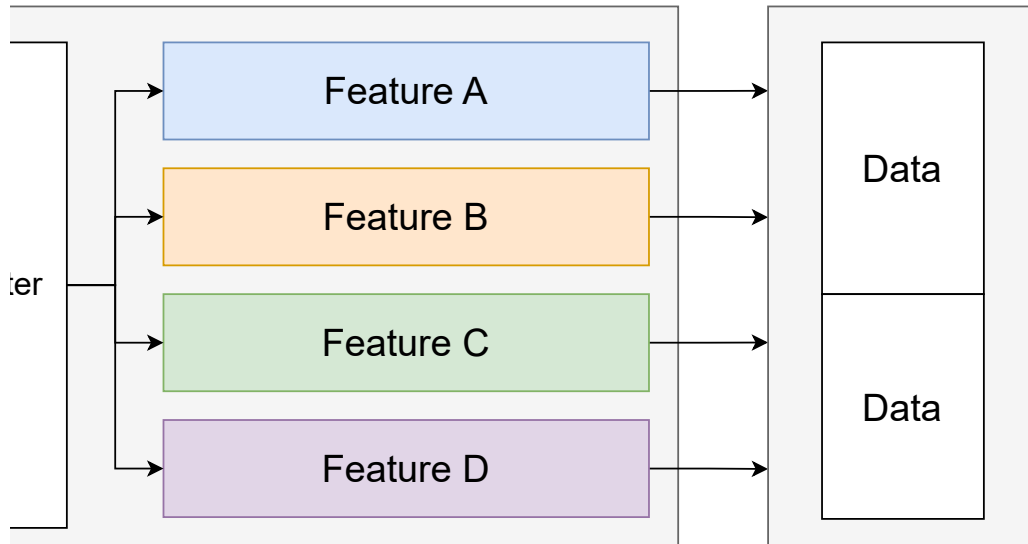
Auth
Middle-
ware → Rout

Les monolith
ac

la majorité des

Serveur monolithique

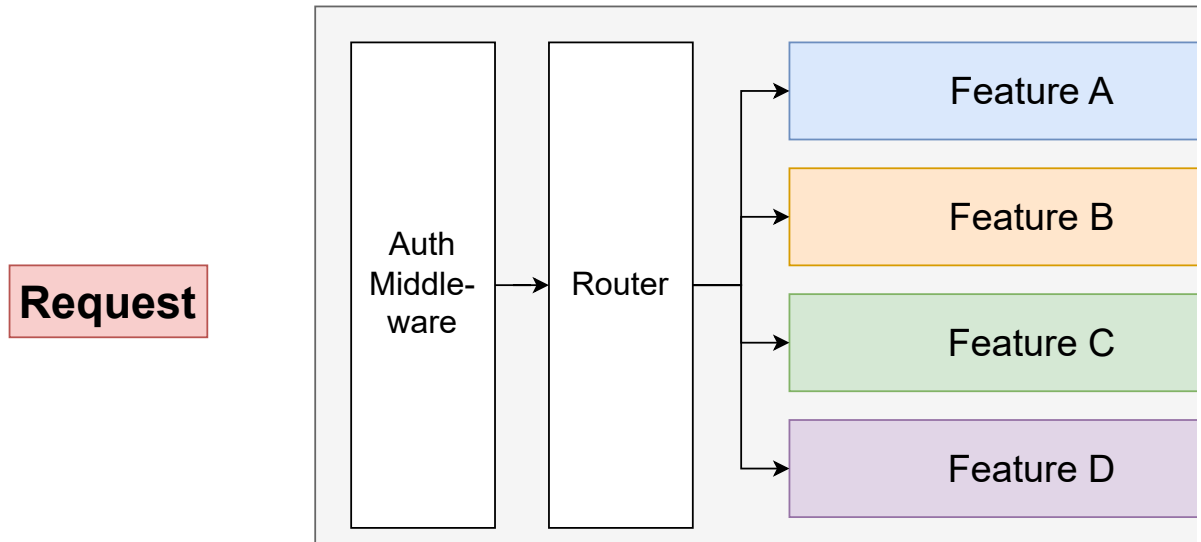
Database



ces peuvent être une décision architecturale
déquate en fonction de la situation.

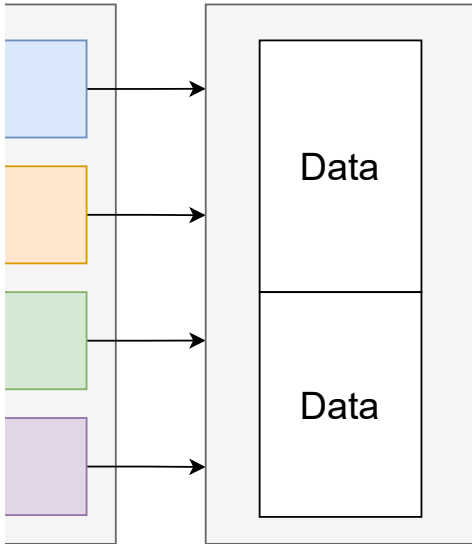
s logiciels ont été développés avant la révolution
du cloud et des microservices.

Serveur monolithique



Un **monolithe** est une application complète dans un seul processus.

Database



te qui s'exécute dans

Scalabilité

Team Devs

Désordre

Utilisateurs

Maintenances

Utilisation d'un seul langage de Programmation

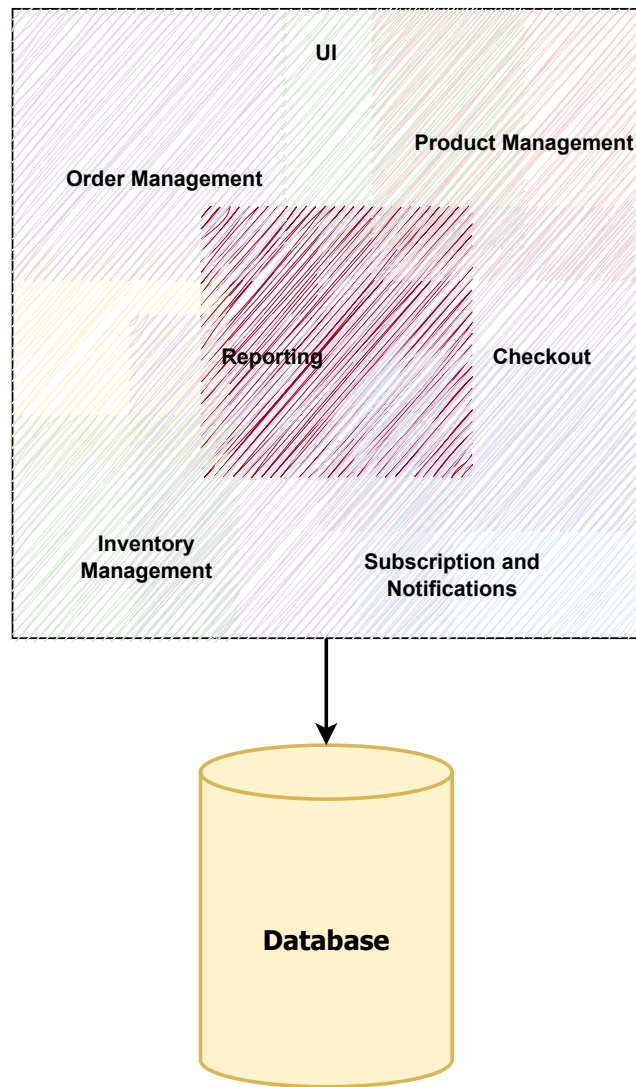
Difficulté de déploiement

Risque d'arrêt

Scalabilité Verticale coûteuse

**Réduction de l'agilité sur les technologies même
l'équipe**

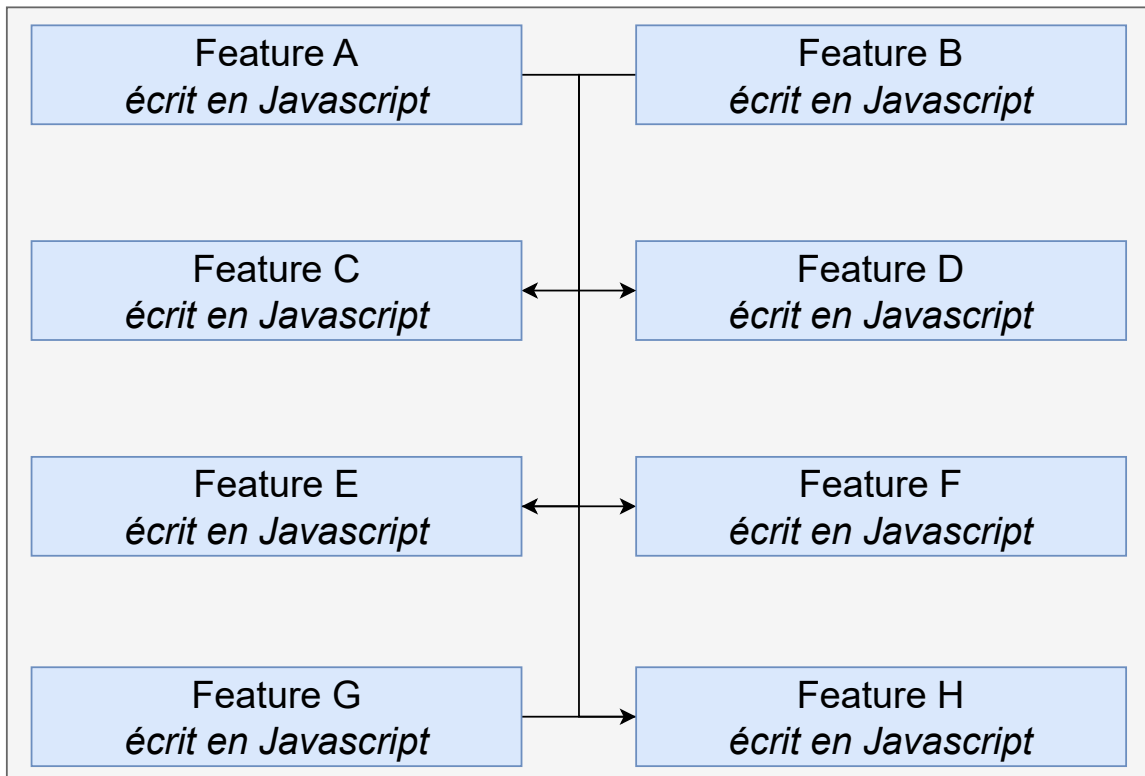
Plusieurs autres PB....

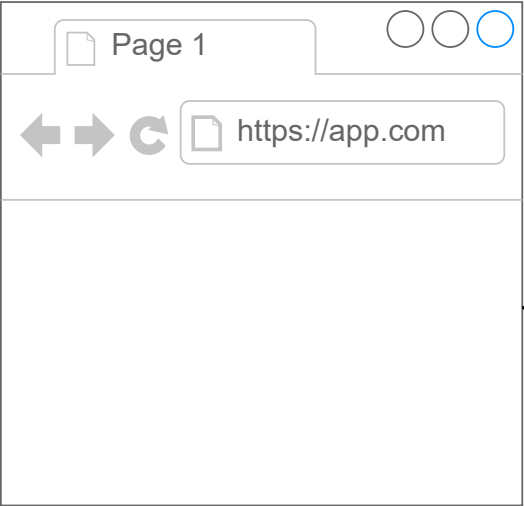


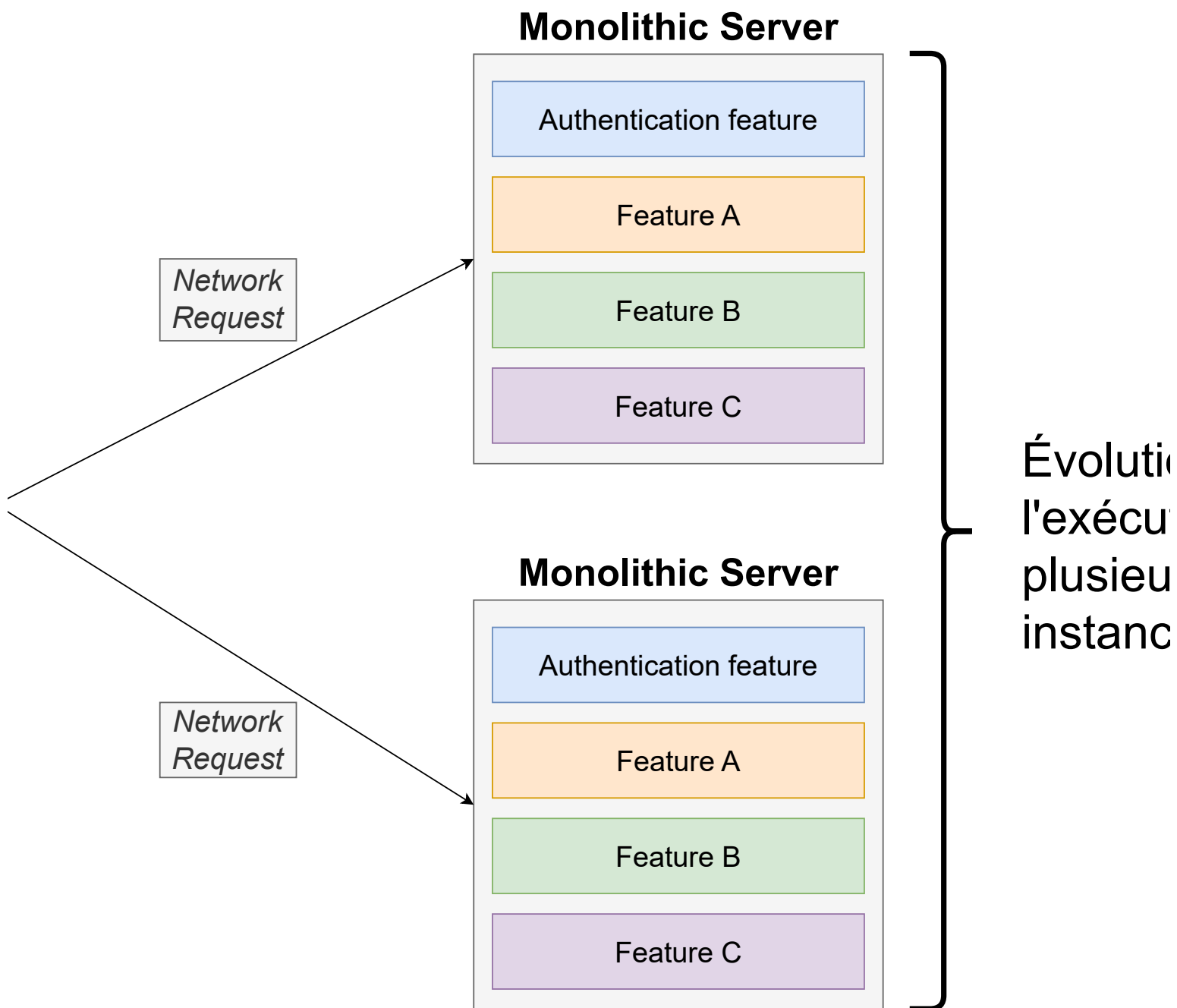
LE COMMERCE ÉLECTRONIQUE EST UN MONOLITHE
PATCHWORK DONT LES FRONTIÈRES ENTRE LES
DIFFÉRENTS DOMAINES NE SONT PAS CLAIRES

Plusieurs autres PB....

Serveur monolithique



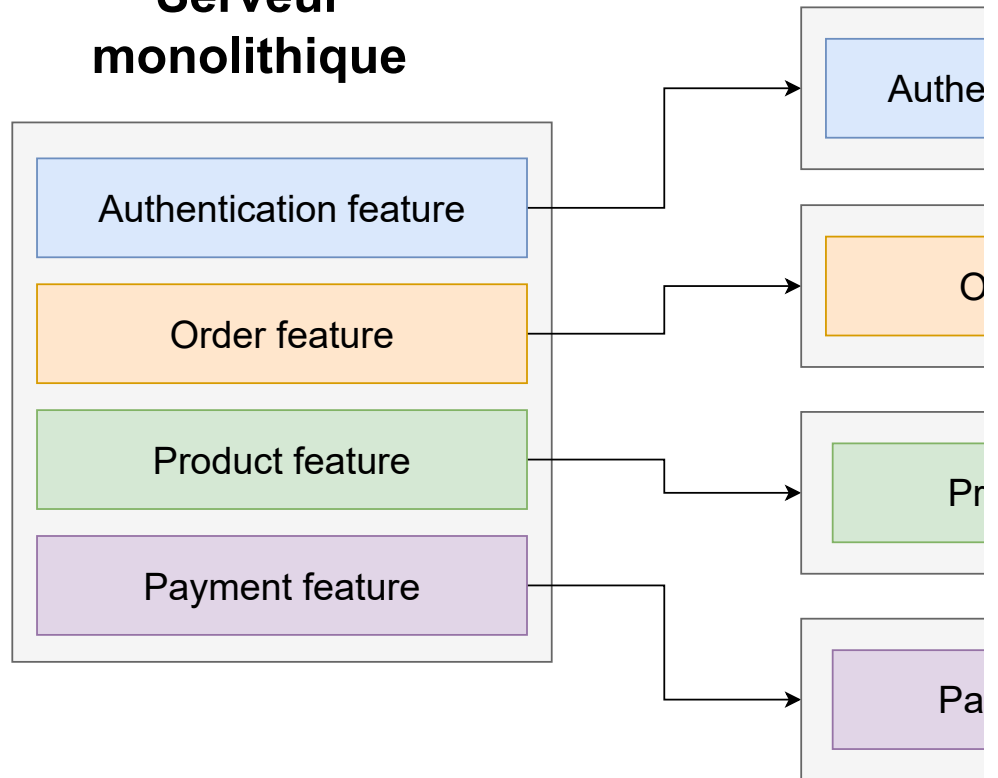


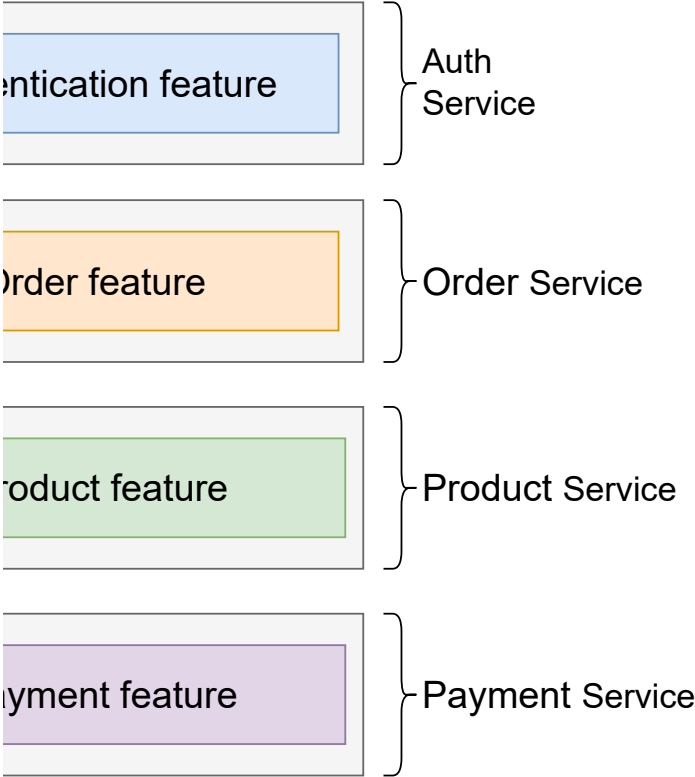


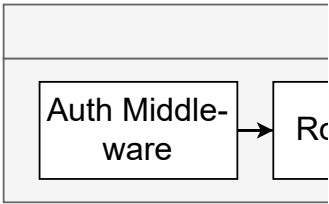
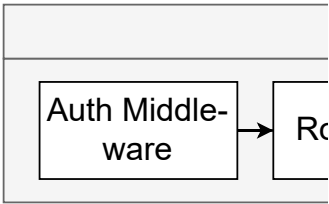
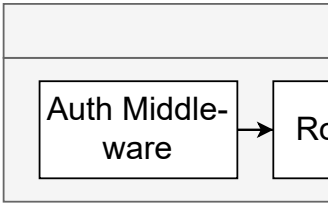
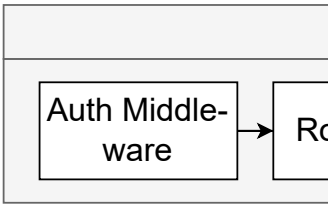
on par
tion de
irs
es

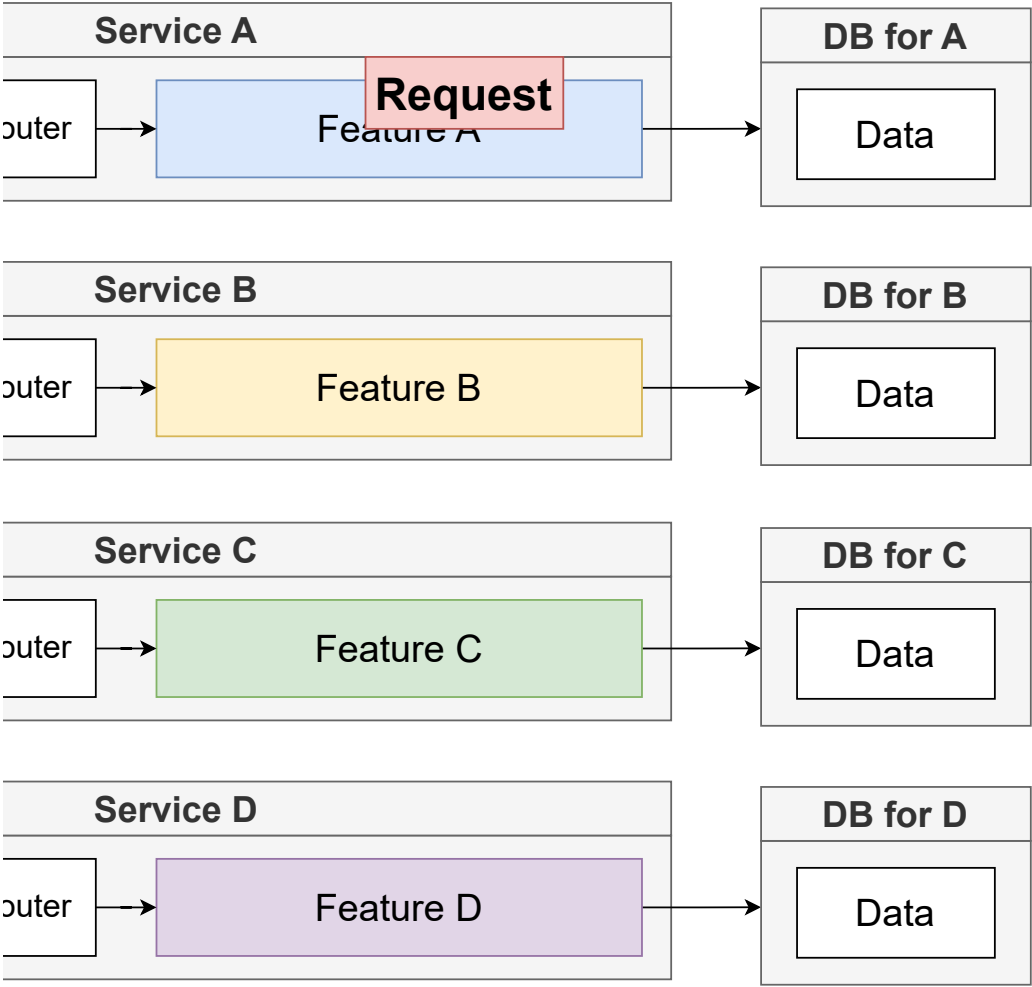
Microservices

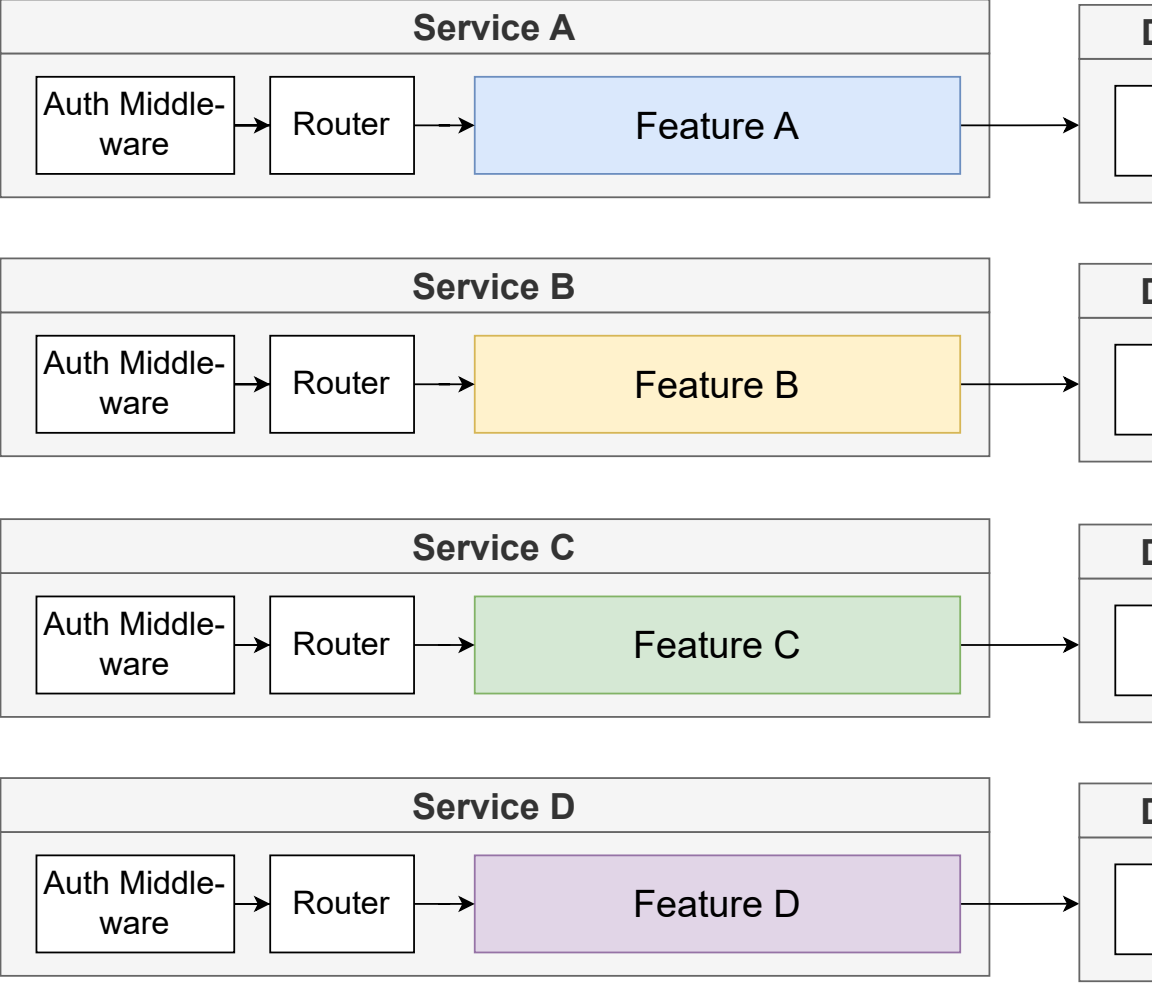
Serveur monolithique

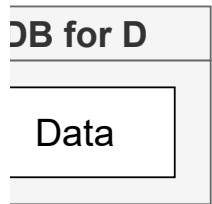
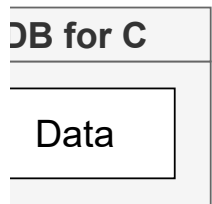
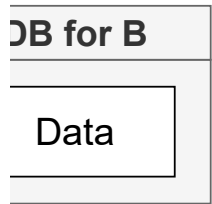
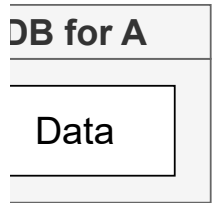




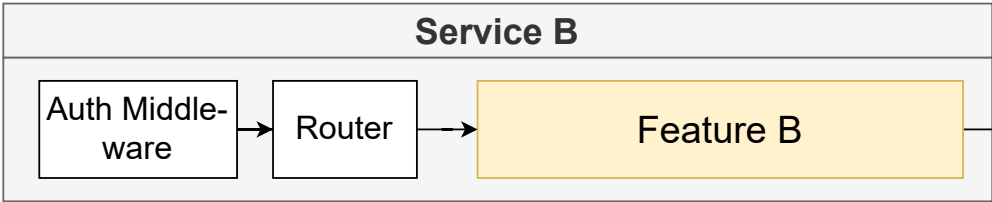
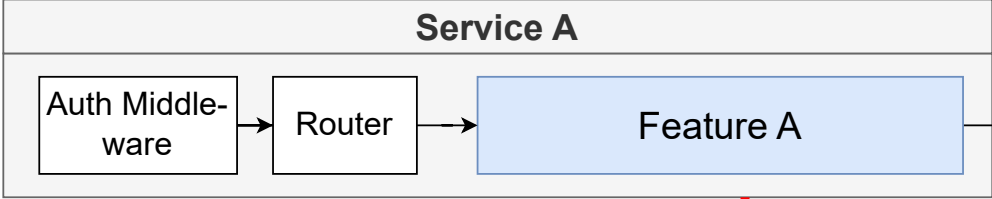


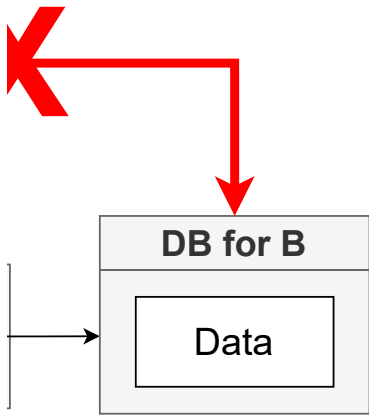
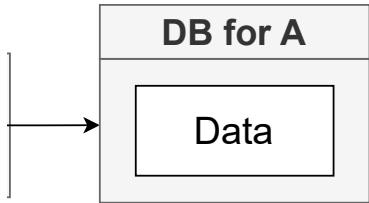






Chaque service dispose de sa propre base de données (s'il en a besoin).





Les services n'accèdent jamais à la base de données d'un autre service.

**Chaque service
a sa propre base de données
a besoin**

**Les services n'ont pas accès
à la base de données
service**

**e dispose de sa
données (s'il en
oin).**

**ccèdent jamais
nées d'un autre
ice.**

D'accord, mais
pourquoi ?

Pourquoi une base de données

Nous voulons que chaque service fonctionne indépendamment des autres services.

Le schéma ou la structure de la base de données est définie à l'avance et ne change pas de manière inattendue.

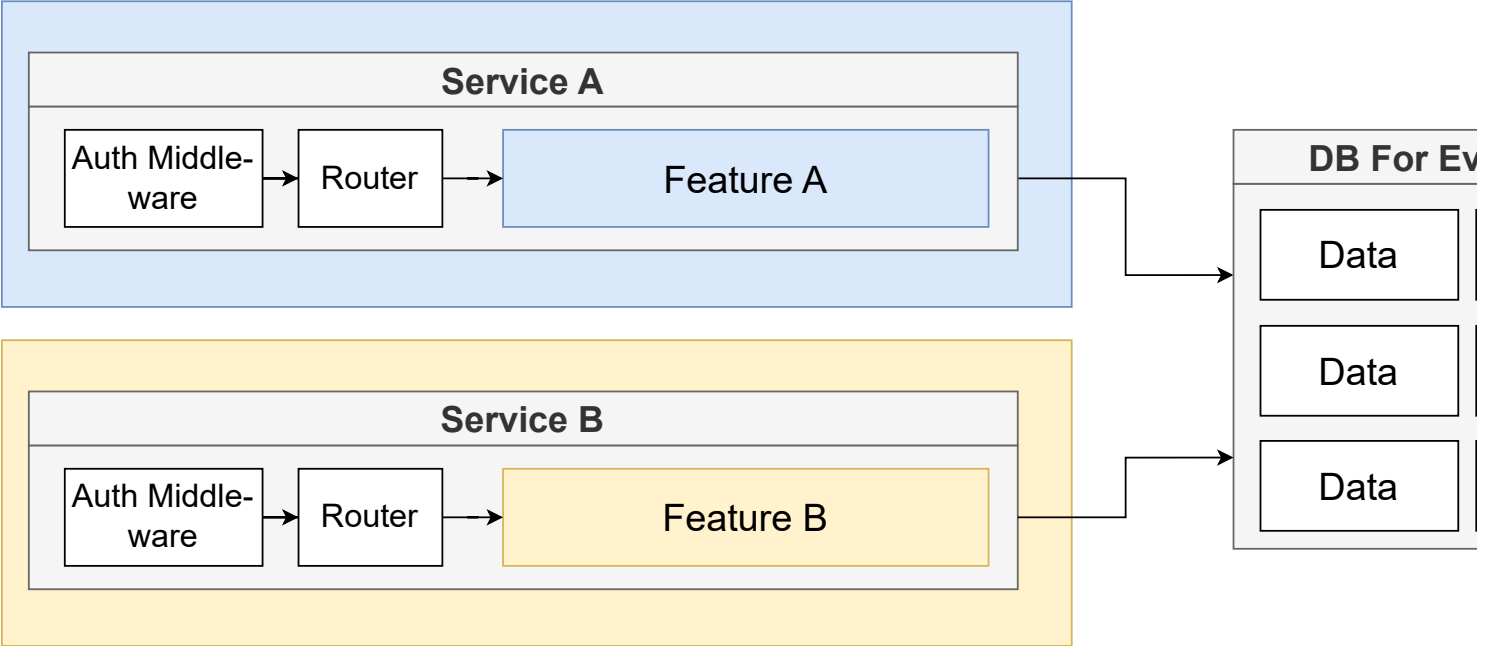
Certains services peuvent fonctionner plus efficacement en utilisant une base de données (sql ou no sql).

ées par service ?

lémentamment des autres

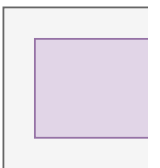
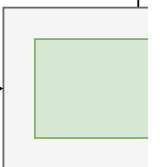
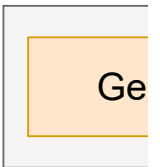
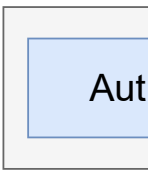
s peut changer de manière

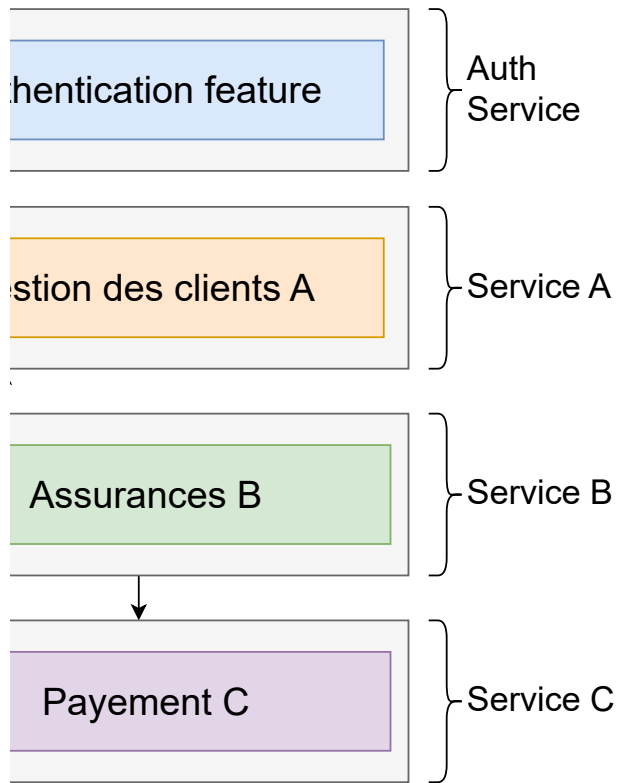
ement avec différents types
nosql).

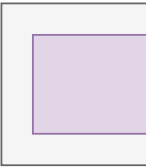
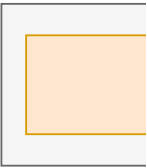
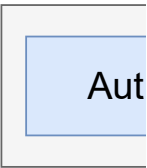


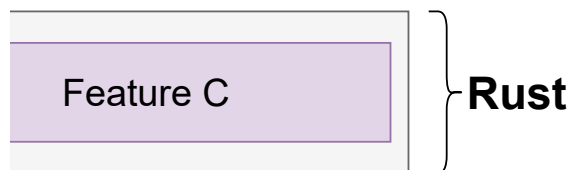
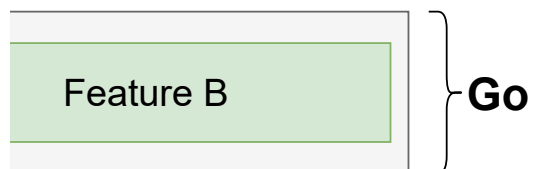
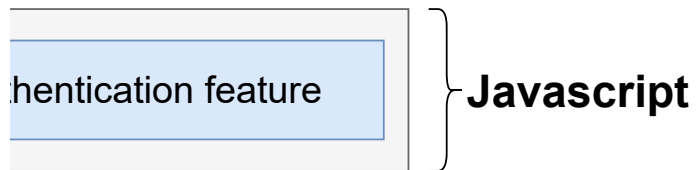
| Everything |
|------------|
| Data |
| Data |
| Data |

Besoin d'apporter une modification
à l'article B ? Nous savons
exactement où aller !









Choisir la meilleure
langue pour le travail

Un microservice unique contient

Routing

Middlewares

Business
Logic

Database
access

pour mettre en œuvre une
fonctionnalité de notre application

L'architecture microservice est un style architectural qui structure une application comme une collection de services qui sont :

- déployables de manière indépendante
- faiblement couplés
- Organisés autour des capacités de l'entreprise
- Appartenant à une petite équipe

L'architecture microservice permet à une organisation de fournir des **applications** complexes et de grande envergure de manière rapide, fréquente, fiable et durable - une nécessité pour être compétitif et gagner dans le monde d'aujourd'hui.

source : microservices.io

Alors.... quelles sont les grands défis des microservices ?

communication

Sécurité

Logging

Monitoring

déploiement

données

