

Intégration d'un SGBD dans un système d'information

Couches et architectures



Sommaire

- Principes des couches
- Les principales architectures
- A retenir
- Heroku



Principe de couches

Le développement d'un outils informatique de gestion de données, que l'on appel aussi systeme d'information, s'organise autour de trois fonctionnalités et ce quelques soit la technologie utilisée :

- Le stockage : les données du programmes sont stockées quelques part afin de pouvoir être réutilisées.
- La manipulation : Le programme traite les données en fonction du besoin de l'utilisateur.
- La présentation : Le programme propose à l'utilisateur une interface pour accéder au données et au fonctions.

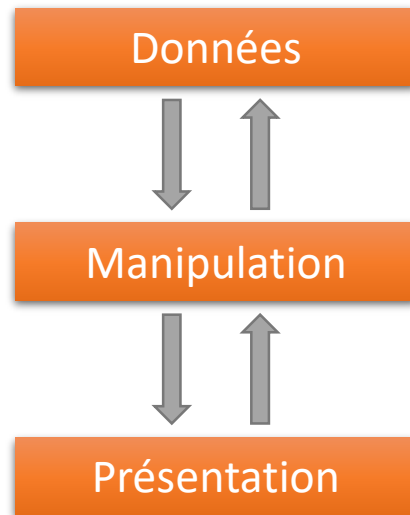
Principes de couches

Exemple : Le projet Netflix

- **MySQL** est le **SGBD** qui est utilisé pour stocker les données.
- Le programme **python** et le framework **Flask** forment la partie du SI qui permet de manipuler les données et exécuter des requêtes SQL.
- La présentation est gérée par le **navigateur web** du client. L'utilisateur pourra sélectionner les informations qu'il souhaite visualiser à l'aide d'un **formulaire**, puis un **tableau de bord** lui sera affiché.

Principes de couches

Chacune de ces fonctionnalités est vue comme une couche qui interagies avec les autres suivant le schéma ci-dessous.



Cette organisation en couche peut :

- Être organisée à l'intérieur d'un seul et unique programme
- Se trouver dans différents outils d'une solution globale. Dans ce cas, chaque couche est un outil.

Les principales architectures

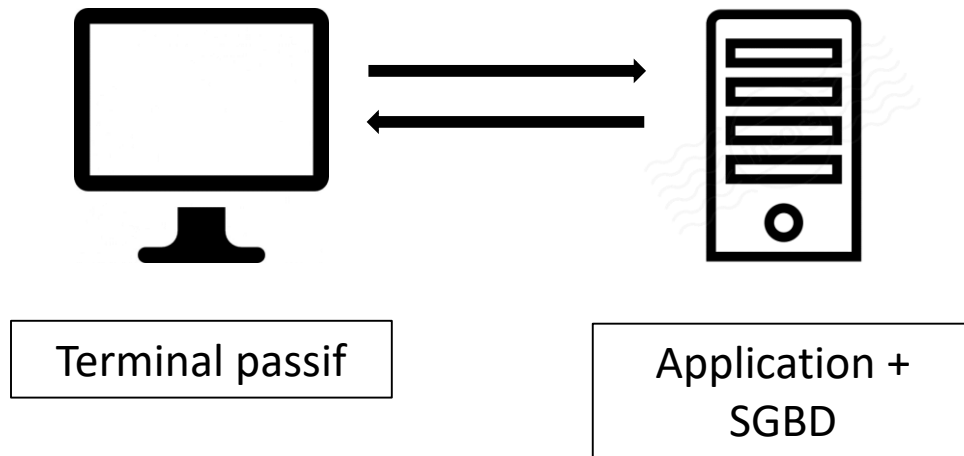
En tenant compte de ces principes, il est possible de définir différentes architectures des composant d'un système.

- Architecture centralisée
- Architecture client serveur
- Architecture 3-tiers
- Architecture client multiseveurs
- Architecture répartie

Les principales architectures

Architecture centralisée :

Cette architecture repose sur un terminal passif et un serveur très puissant connecté en réseau.



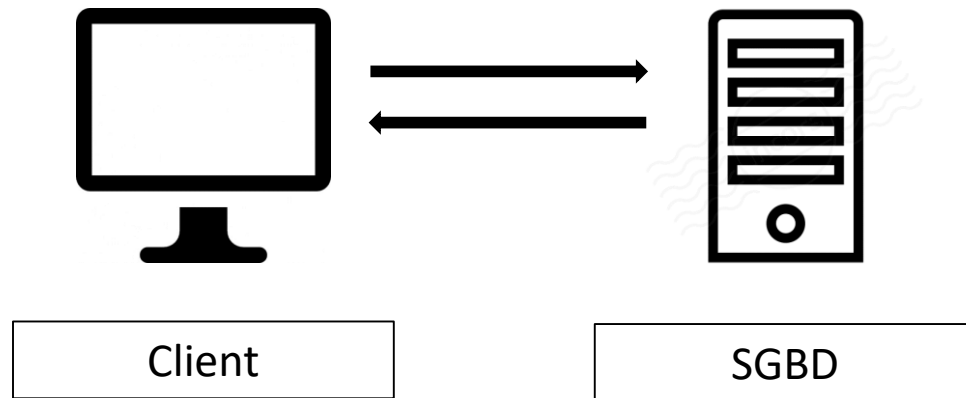
Dans cette organisation, le serveur assure l'exécution de tous les programmes dont le SGBD.

Ce type d'architecture est de moins en moins utilisé.

Toutefois elle semble retrouver un second souffle avec le cloud computing.

Architecture client serveur

Cette architecture repose sur un ou plusieurs clients applicatif et un SGBD connecté en réseau.

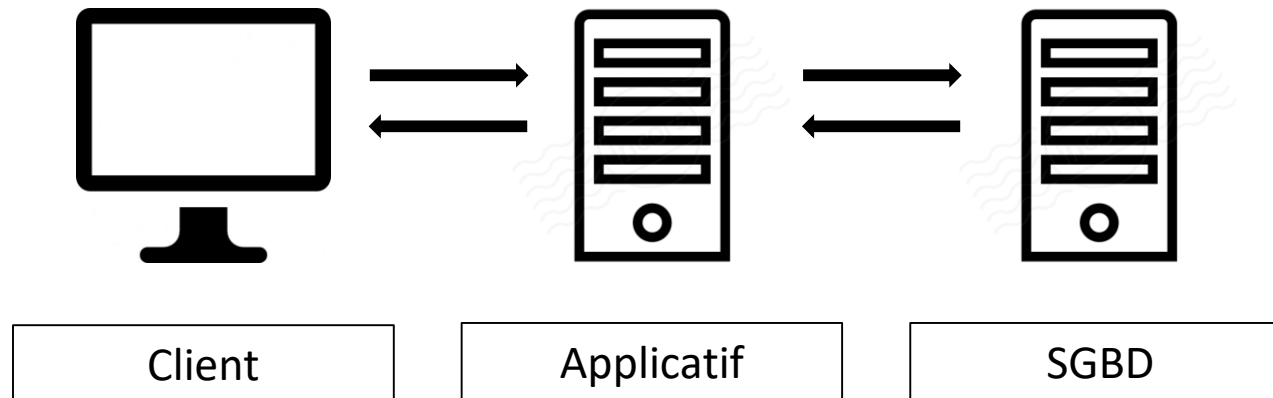


Dans cette organisation le client est considéré comme intelligent car c'est lui qui exécute le programme.

Le SGBD est utilisé pour stocker et centraliser les données.

Architecture 3-tiers

Cette architecture repose sur un ou plusieurs clients, un serveur applicatif et un SGBD, connecté en réseau. Il s'agit de l'architecture la plus répandue.

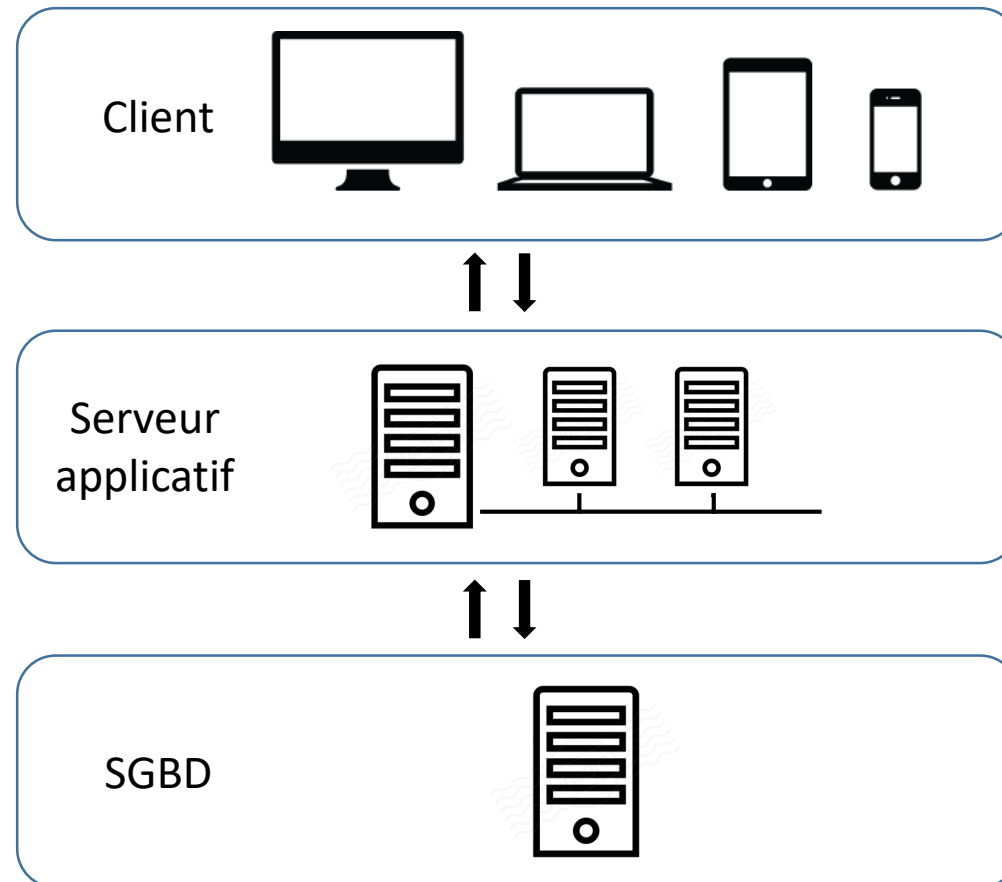


Dans cette architecture, les clients dialoguent avec le serveur applicatif qui assure le traitement métier des données.

Le SGBD assure la gestion des données

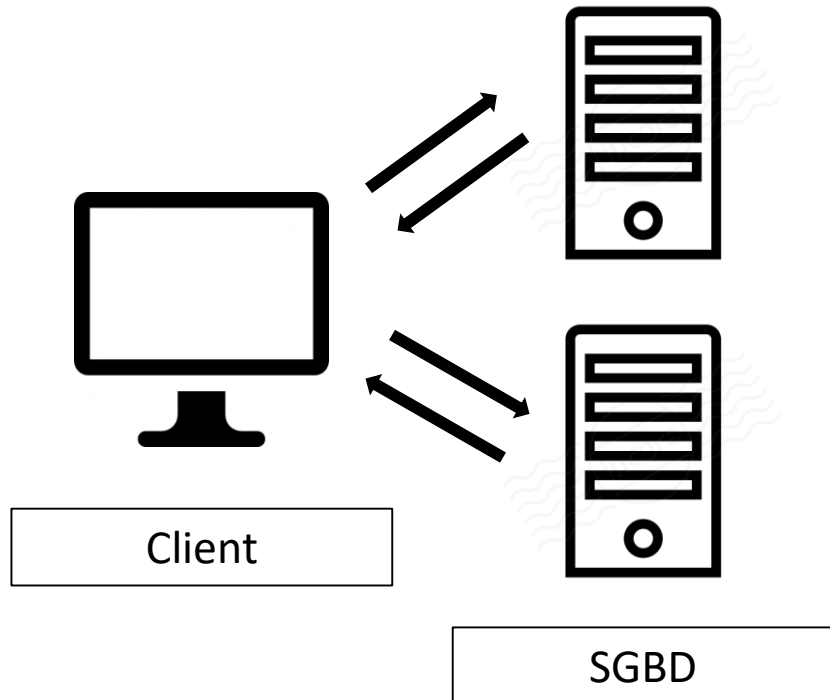
Suivant la taille de l'application, la couche applicative peut contenir plusieurs serveurs afin de répondre de manière plus efficace au clients.

Dans ce cas un système de répartition de charge organise le routage des connexions entre les différents serveurs.



Architecture client multiserveurs

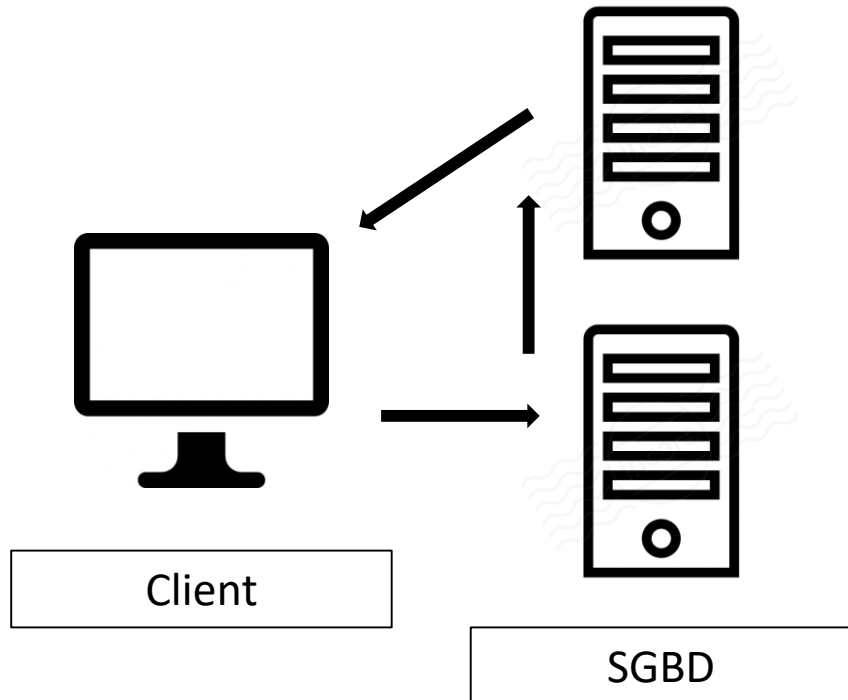
Cette architecture repose sur un ou plusieurs clients et plusieurs SGBD connectés en réseau.



Cette organisation nécessite des clients intelligents capables de s'adresser à plusieurs SGBD et à combiner plusieurs résultats.

Architecture répartie

Cette architecture repose sur un ou plusieurs client et plusieurs SGBD connecté en réseau.



Dans cette organisation les clients ne s'adresse qu'à un seul SGBD.

Le système de gestion des serveurs SGBD répartie les requêtes sur les différents serveurs qui répondent directement au client.

A retenir



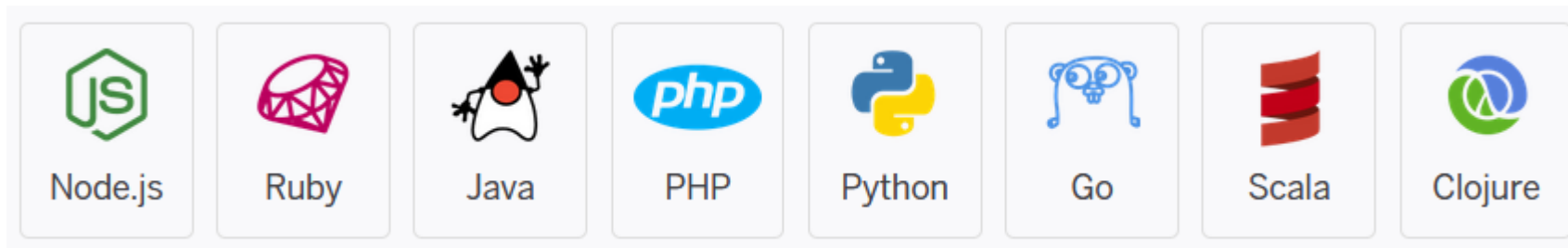
- La répartition des fonctions dans un système d'information permet de définir des composants dans une architecture.
- Dans toute les architecture le SGBD est un composant de base indispensable.
- L'architecture 3-tiers est la plus répandue; elle sépare le SGBD, le serveur applicatif et la présentation du système d'information.

Le choix d'une architecture dépend principalement du système d'information et des contraintes d'exploitation.

Il est stratégique pour l'exploitation de l'application et doit être « pensé » avant la conception et la mise en œuvre.



- C'est un service PaaS (Platform as a Service) donnant accès à distance à des services (gratuits ou payants) :
 - d'hébergement
 - de noms de domaines
 - de certificat SSL
 - de langage de programmation
 - de bases de données.



- Cette solution de cloud computing : utilisation de serveurs distants sur Internet
- L'utilisateur dispose d'une infrastructure de développement maintenue par un prestataire externe et peut se concentrer uniquement sur ses applications.
- Inconvénient : Seul PostgreSQL et MongoDB sont pris en charge.

<https://www.youtube.com/watch?v=Uj8-m35Uzm0>

<https://openclassrooms.com/fr/courses/4425066-concevez-un-site-avec-flask/4526644-deployez-efficacement-sur-le-cloud-avec-heroku>

<https://www.youtube.com/watch?v=Li0Abz-KT78>