



# Bases de données

420-1B2-SW Développement des bases de données



# Bases de données

- Depuis près de 50 ans, la recherche dans le domaine de la **base de données** a eu un profond impact sur l'économie et la société, alimentant une industrie aujourd'hui évaluée à plus 50 milliards US.

- La **base de données** pilote l'ensemble des projets d'aujourd'hui...

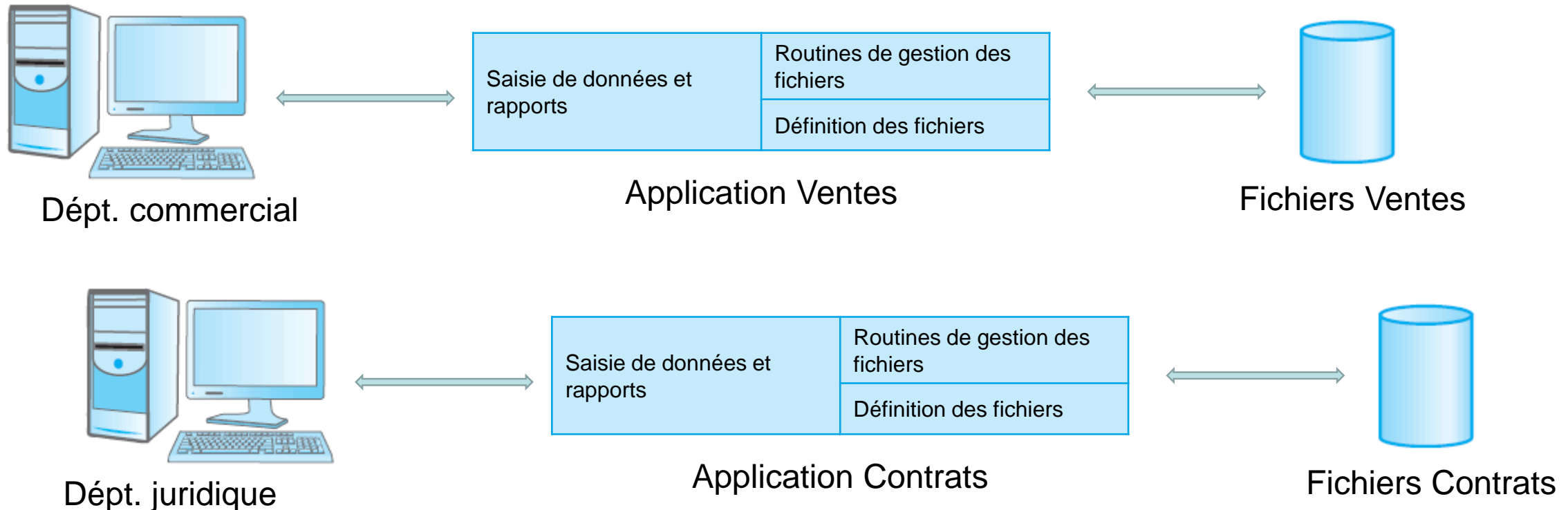




# Bases de données

– Traditionnellement, les données sont :

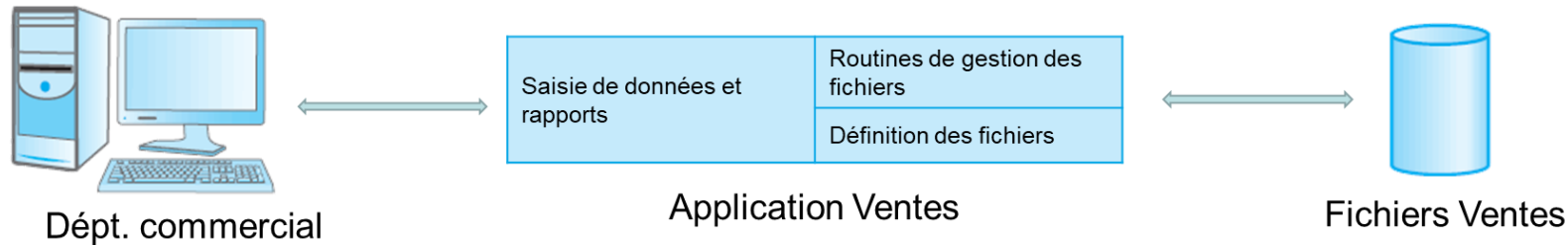
- Stockées dans des fichiers;
- Traitées par des logiciels dédiés.





# Bases de données

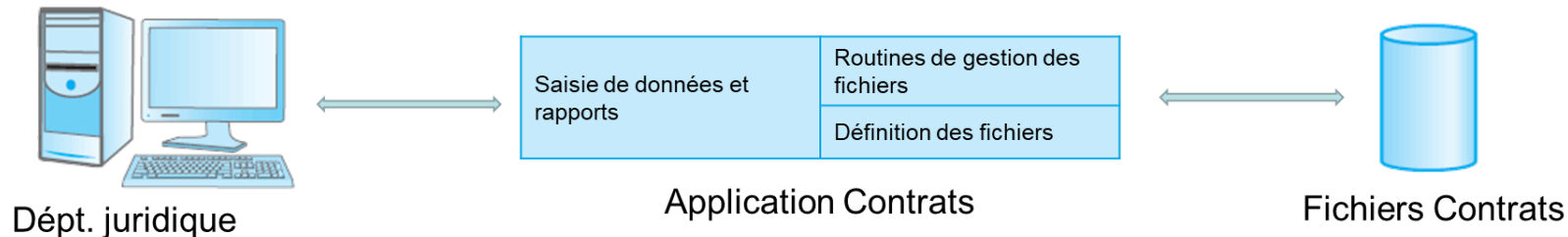
## – Exemples :



**Propriété**(numPropriété, rue, ville, codePostal, type, pièces, location, numPropriétaire)

**Propriétaire**(numPropriétaire, nom, prénom, téléphone)

**Client**(numClient, nom, prénom, téléphone, typePréf, locMax)



**Bail**(numBail, numPropriété, numClient, location, modePaiement, caution, début, fin, durée)

**Propriété**(numPropriété, rue, ville, codePostal, location)

**Client**(numClient, nom, prénom, téléphone)



# Bases de données

- Limitations de l'approche orientée fichiers :
  - Séparation et isolation des données
    - Lorsque les données sont isolées dans différents fichiers, les informations qui devraient être accessibles sont difficiles à atteindre.
    - Par exemple, pour produire une liste des propriétés qui pourraient répondre aux attentes de clients, il est nécessaire de créer un fichier temporaire et d'y stocker une à une les entrées du fichier *Propriété* dont *type* équivaut à *typePréf* du *Client* et dont *location* est inférieur ou égal à *locMax* du *Client*.
  - Duplication des données
    - Dans le précédent exemple, le *nom*, *prénom* et *téléphone* du *Client* sont stockés à deux endroits différents. Une modification dans l'un des fichiers ne sera pas reflétée dans l'autre fichier, avec les inconvénients et risques d'erreur qui en découlent.



# Bases de données

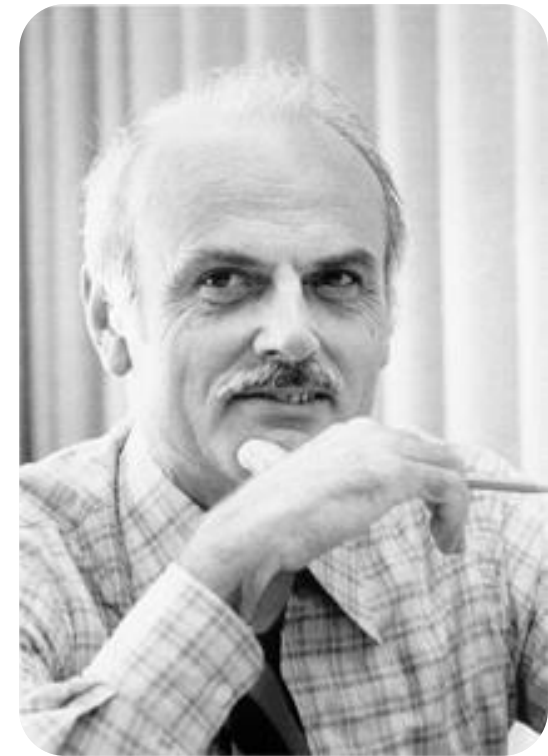
- Limitations de l'approche orientée fichiers (*suite*) :
  - Dépendance des données
    - La structure des fichiers de données est stockée dans le code.
    - Par exemple, pour modifier la taille de *adresse* de 40 à 45 caractères, il est nécessaire de coder une routine unique qui créera un fichier temporaire afin d'y transférer l'ensemble des entrées altérées.
  - Formats incompatibles
    - De par leur façon respective de traiter leurs types de données, le fichier généré par un code COBOL n'est pas forcément accessible par un code C.
  - Requêtes fixes stockées dans les programmes
    - Les logiques implémentées dans un code permettant de requérir et d'agrégier les données ne sont pas réutilisables dans les autres programmes.





# Bases de données

- Dans les années 1960, **Edgar Frank Codd** œuvre chez **IBM** et cherche une solution satisfaisante à l'arrangement des données. Cette recherche aboutit au **modèle relationnel**.
  - En 1969, un rapport technique chez IBM décrit le modèle. Le modèle fait l'objet d'une publication en **1970** :  
*"A Relational Model of Data for Large Shared Data Banks"*  
CACM 13, No. 6, June 1970.

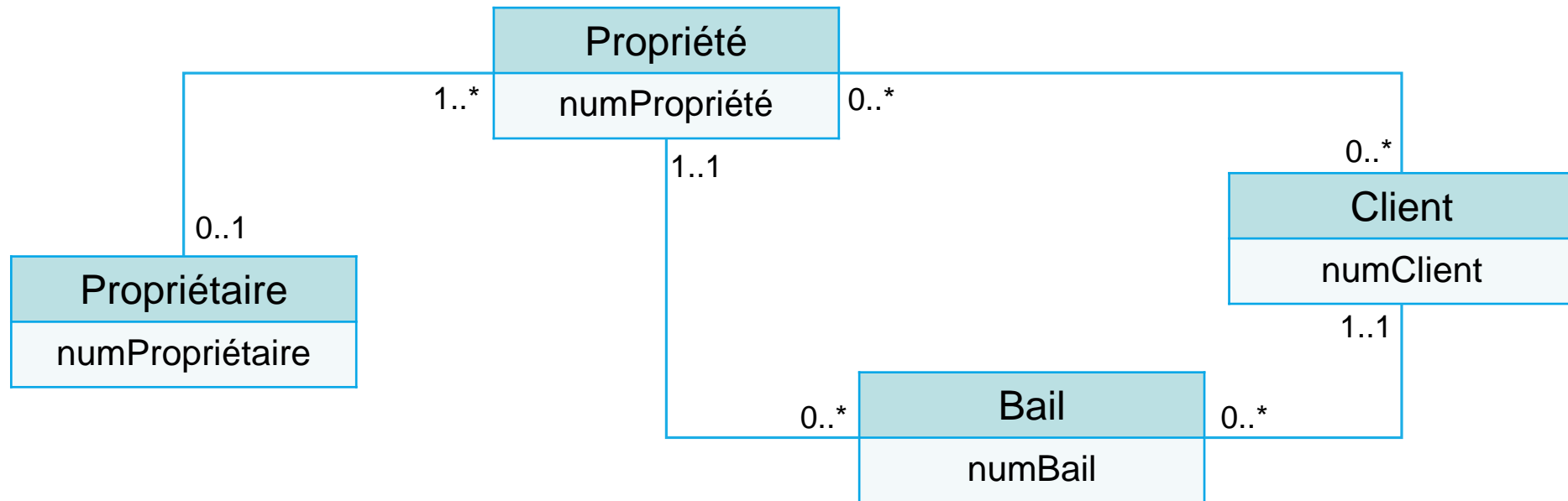


Edgar Frank Codd



# Modèle relationnel

- Le **modèle relationnel** (aussi : modèle entités-relations) propose la définition d'entités par les attributs qui les composent ainsi que des relations définissant leurs liens logiques :

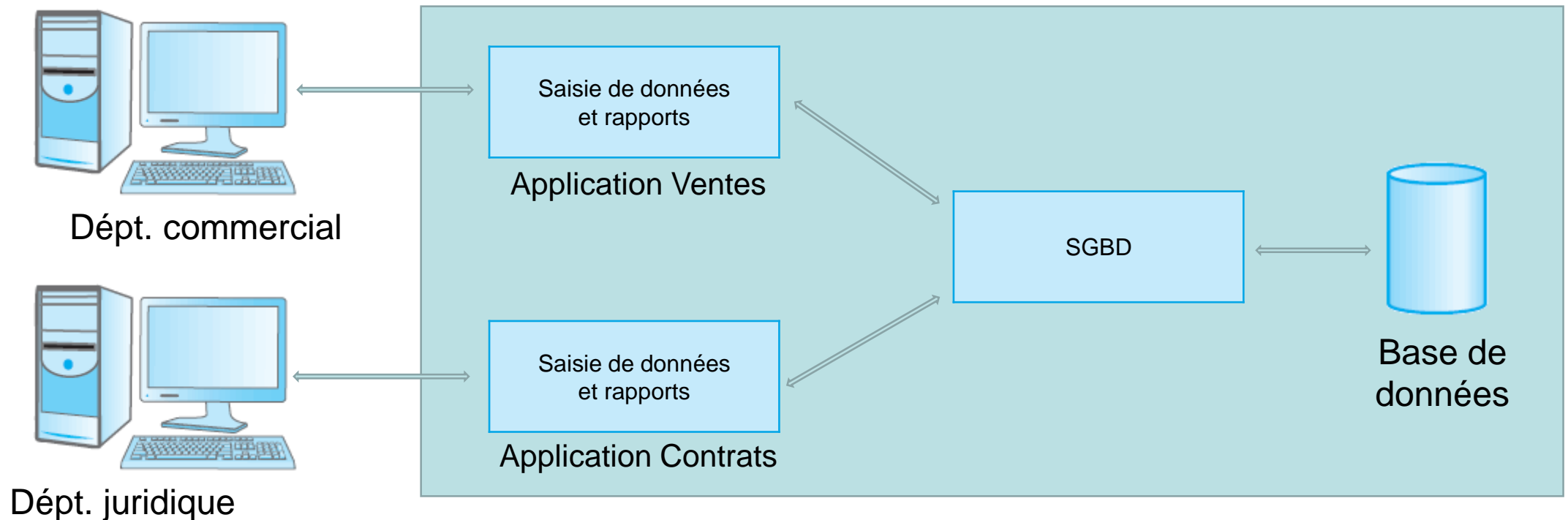


Remarquez qu'il n'y a ici aucune considération quant au "où" ni au "comment" les données sont stockées.



# Système de gestion des bases de données

– Le système de gestion des bases de données (SGBD) :



**Propriété**(numPropriété, rue, ville, codePostal, type, pièces, location, numPropriétaire)

**Propriétaire**(numPropriétaire, nom, prénom, téléphone)

**Client**(numClient, nom, prénom, téléphone, typePréf, locMax)

**Bail**(numBail, numPropriété, numClient, location, modePaiement, caution, début, fin, durée)

# Système de gestion des bases de données

- En 1977, le jeune Larry Ellison âgé de 33 ans a lu les écrits de Codd et commence à implanter le modèle relationnel...



Lawrence Joseph Ellison. Fortune estimée à 69 milliards US\$ en 2019.



# Système de gestion de base de données

420-1B2-SW Développement des bases de données

# Système de gestion de base de données

Temps	Développement
→ années 60	Systemes orientés fichiers
60s	Modèles hiérarchiques et réseau
70s	<b>Modèle relationnel</b> proposé
1979	Implémentation modèle relationnel
1987	Standardisation ISO <b>SQL</b>
1990s	Entrepôts de données
2010s	Infonuagique







# Système de gestion de base de données

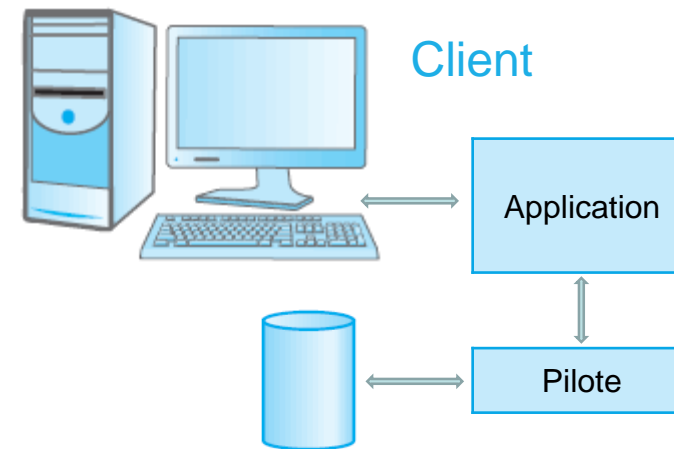
- Quelques **avantages** du système de gestion de base de données :
  - Diminue la redondance des données
  - **Intégrité des données** Les SGBD exposent des règles d'intégrité référentielle et autres contraintes, unicité, format, etc.
  - **Sécurité** Les serveurs SGBD procèdent à l'authentification et à l'assignation des permissions et autorisations associées au compte ou - dans certains cas - au rôle, groupe.
  - **Simplifie l'accès concurrentiel** La majorité des SGBD prévoient des mécanismes permettant à plusieurs utilisateurs d'accéder en simultané aux mêmes données.
  - *etc.*

# Système de gestion de base de données

– Il existe deux types de systèmes de gestion de base de données relationnels :

- SGBDR **fichier** comme Access, dBase ou Paradox.

Le client accède au fichier de données par le biais d'un pilote local et procède localement au traitement des enregistrements désirés.

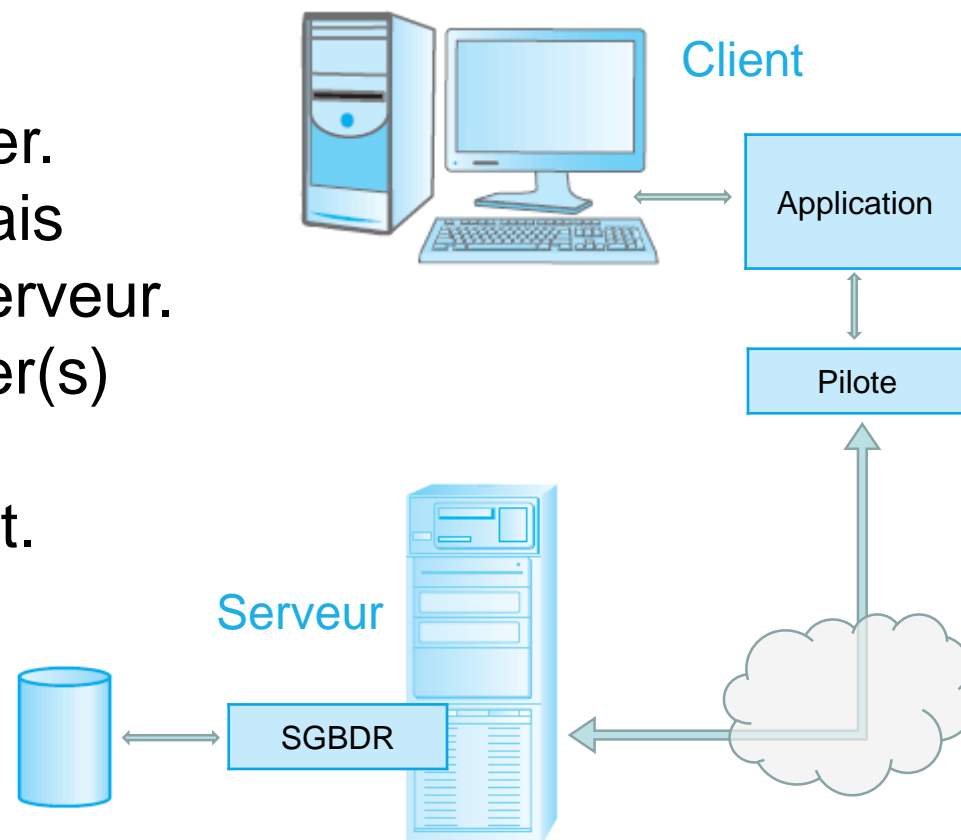




# Système de gestion de base de données

– Il existe deux types de systèmes de gestion de base de données relationnels :

- **Serveur** SGBDR comme Oracle, MySQL ou SQL Server. Le client doit passer par le biais d'un service exécuté par le serveur. Le service accède au(x) fichier(s) et procède au traitement des données désirées par le client.



## – Rôles dans un environnement de base de données



### **Administrateur de base de données (DBA)**

Assure la sécurité, l'intégrité et la haute-disponibilité de l'infrastructure de données (permissions, sauvegardes, chiffrement, répartition de la charge, réplication, etc).



### **Utilisateur**

Utilisateurs naïfs. Certains plus sophistiqués savent connecter une base de données avec Excel par exemple. Certains voudront même vous montrer comment créer une BD Access.



### **Designer de base de données**

Design les tables, relations, vues et autres objets de données. Rôle souvent accompli par l'analyste ou un développeur senior.



### **Développeur**

Développe souvent avec les instructions SQL mais utilise de plus en plus des plateformes ORM (object-relational mapping) apportant une abstraction du SGBD sous-jacent.



## – Références

- CONNOLLY, T. et BEGG, C. *"Database Systems: A Practical Approach to Design, Implementation, and Management, Global Edition"*, Editions Pearson (2015) ISBN: 9781292061184.
- CODD, E.F. *"A Relational Model of Data for Large Shared Data Banks"*, CACM 13, No. 6, June 1970.
- db-engines.com, consulté en décembre 2019.
- wikipedia.org, consulté en décembre 2019.