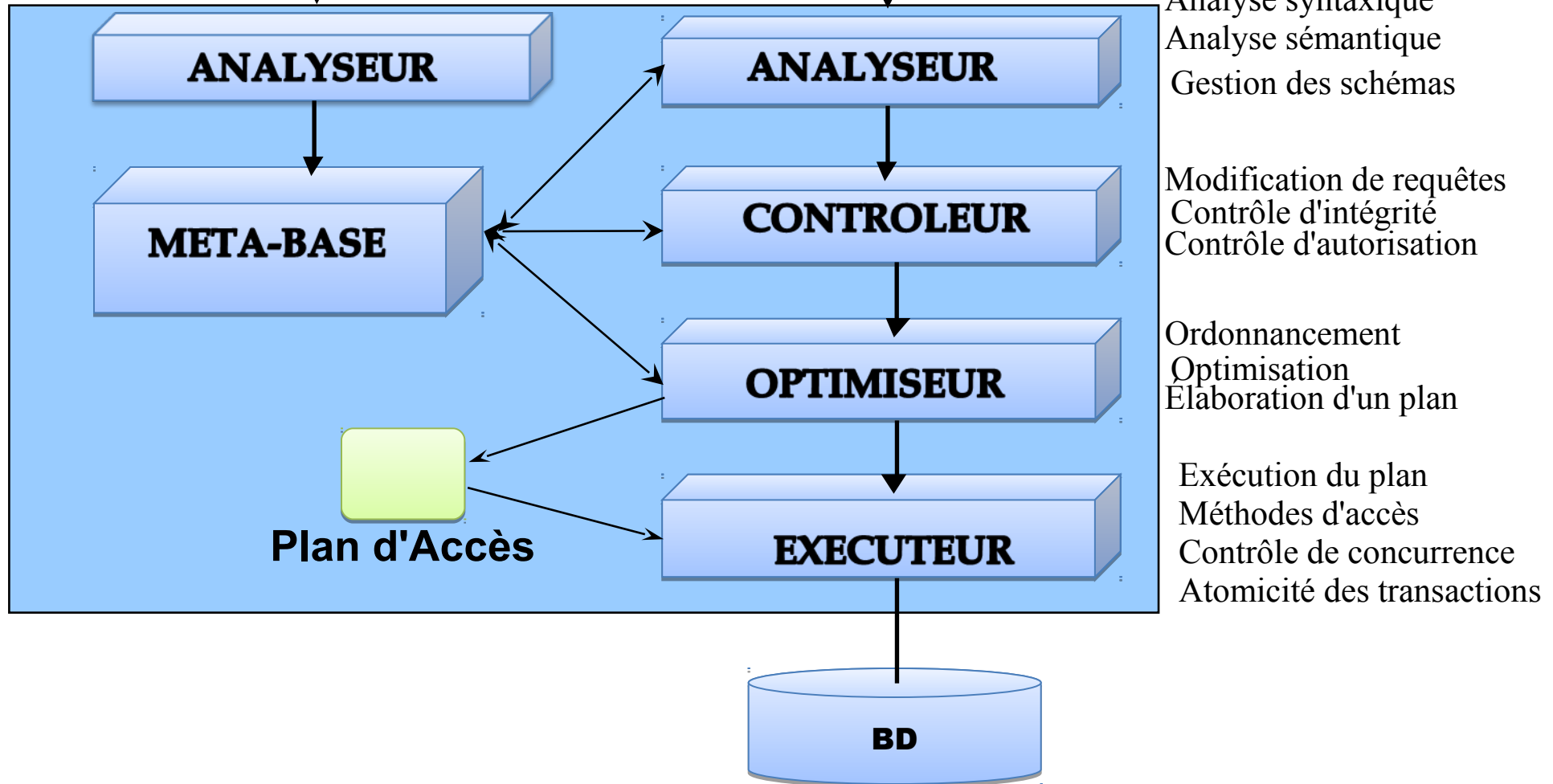


Fonctionnalités des SGBD

Architecture d'un SGBD

SQL LDD

SQL LMD



1. Description des données

- **SQL LDD**

- L'analyseur vérifie la correction des commandes
- Il stocke le schéma dans la méta-base (catalogue système)
- Celle-ci est souvent une BD relationnelle

- **Organisation de la méta-base**

- C'est un ensemble de tables décrivant les autres (et elle même)
- Bases, Tables, Attributs, Domaines, Clés, ...
- Vues, Contraintes, Index, ...

- **Commandes traitées**

- Create Table, Alter Table, etc.

Noyau de métabase

- **SCHEMAS (CATALOG, NOMB, Créateur, Caractère_Set, ...)**
- **TABLES (CATALOG, NOMB, NOMR, Type, ...)**
- **DOMAINS (CATALOG, NOMB, NOMD, Type, Défaut, Contrainte, ...)**
- **COLUMNS (CATALOG, NOMB, NOMR, NOMA, Pos, Type, ...)**
- **TYPES (CATALOG, NOMB, NOM, MaxL, Precision, ...)**
- **CONSTRAINTS (CATALOG, NOMB, NOMC, TypeC, NomR, ...)**
- **USERS (NOM, ...)**

2. Manipulation des Données

- **L'analyseur analyse la requête**
 - Analyse syntaxique selon la grammaire SQL
 - Analyse sémantique selon la métabase
 - Traduction en arbre d'algèbre relationnel
- **Syntaxe SQL (rappel)**
 - **Select** <Liste de champs ou de calculs à afficher>
 - **From** <Liste de relations mises en jeu>
 - **Where** <Liste de prédicats à satisfaire>
 - **Group By** <Groupement éventuel sur un ou plusieurs champs>
 - **Order By** <Tri éventuel sur un ou plusieurs champs>
- **Représenter les arbres algébriques des requêtes suivantes**

Exemple de question SQL (1)

- **Nom et description des médicaments de type aspirine**

```
Select      Nom, Description  
From Médicaments  
Where      Type = 'Aspirine'
```

Exemple de question SQL

(2)

- **Patients de Tiko ayant effectués une visite le 15 juin**

Select Patients.Nom, Patients.Prénom
From Patients, Visites
Where Patients.Id-P = Visites.Id-P
and Patients.Ville = 'Tiko'
and Visites.Date = '15 juin'

Exemple de question SQL (3)

- **Dépenses effectuées par patient trié par ordre décroissant**

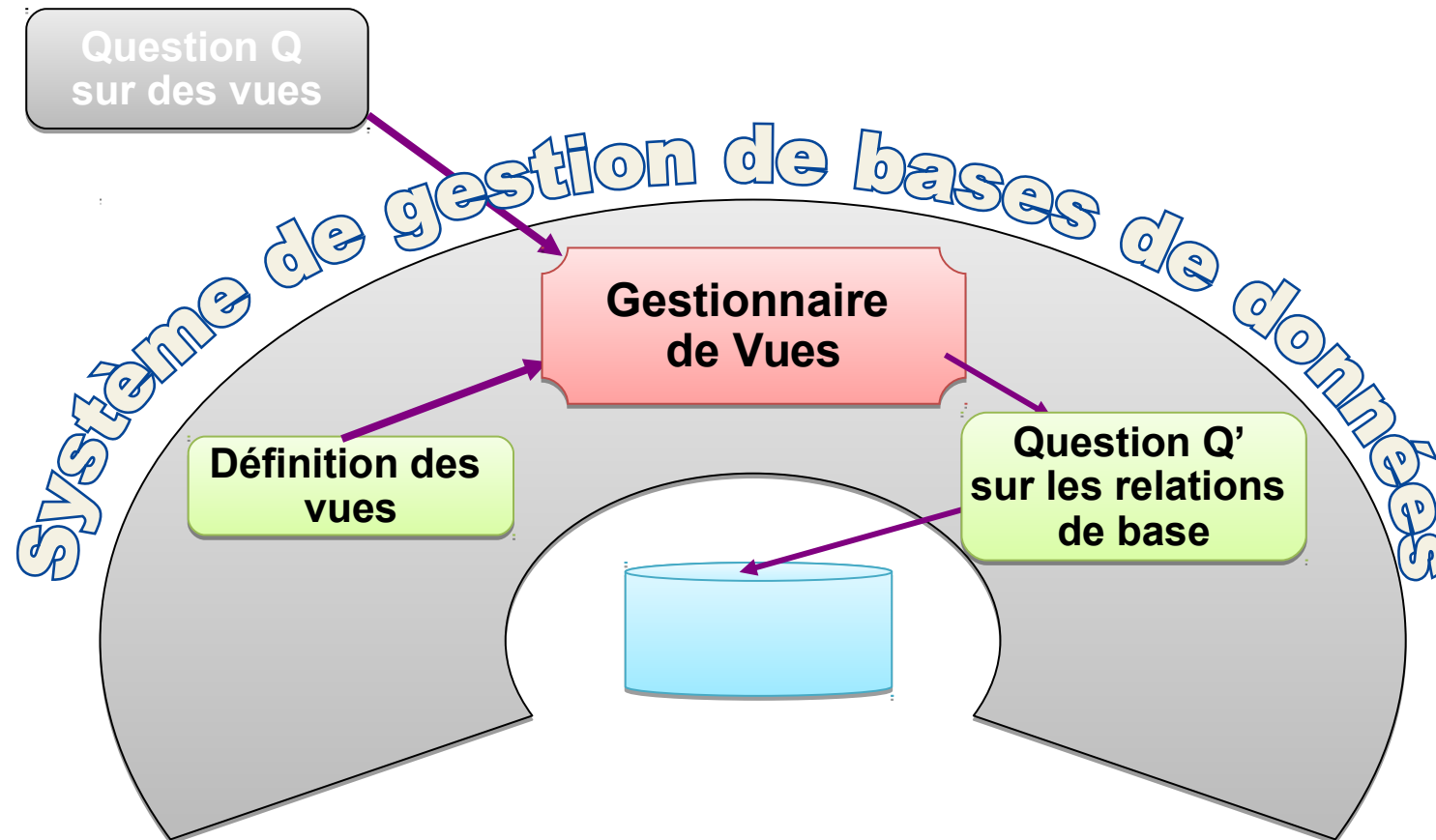
```
Select    Patients.Id-P, Patients.Nom, sum(Prix)
From      Patients, Visites
Where     Patients.Id-P = Visites.Id-P
GroupBy   Patients.Id-P, Patients.Nom
OrderBy sum(Prix) desc
```


3. Gestion des vues

- Les vues permettent d'implémenter l'indépendance logique en permettant de créer des **objets virtuels**
- Vue = Question SQL stockée
- Le SGBD stocke la **définition** et non le résultat
- Exemple : la vue des patients de Yaoundé
Create View Ongola **as** (
Select Nom, Prénom
From Patients
Where Patients.Ville = 'Yaoundé')

Gestion des vues

Le SGBD **transforme** la question sur les vues en question sur les relations de base



Syntaxe et Exemple

- **CREATE VIEW** <nom de table> [(<nom de colonne>+)]

AS <spécification de question> [**WITH CHECK OPTION**]

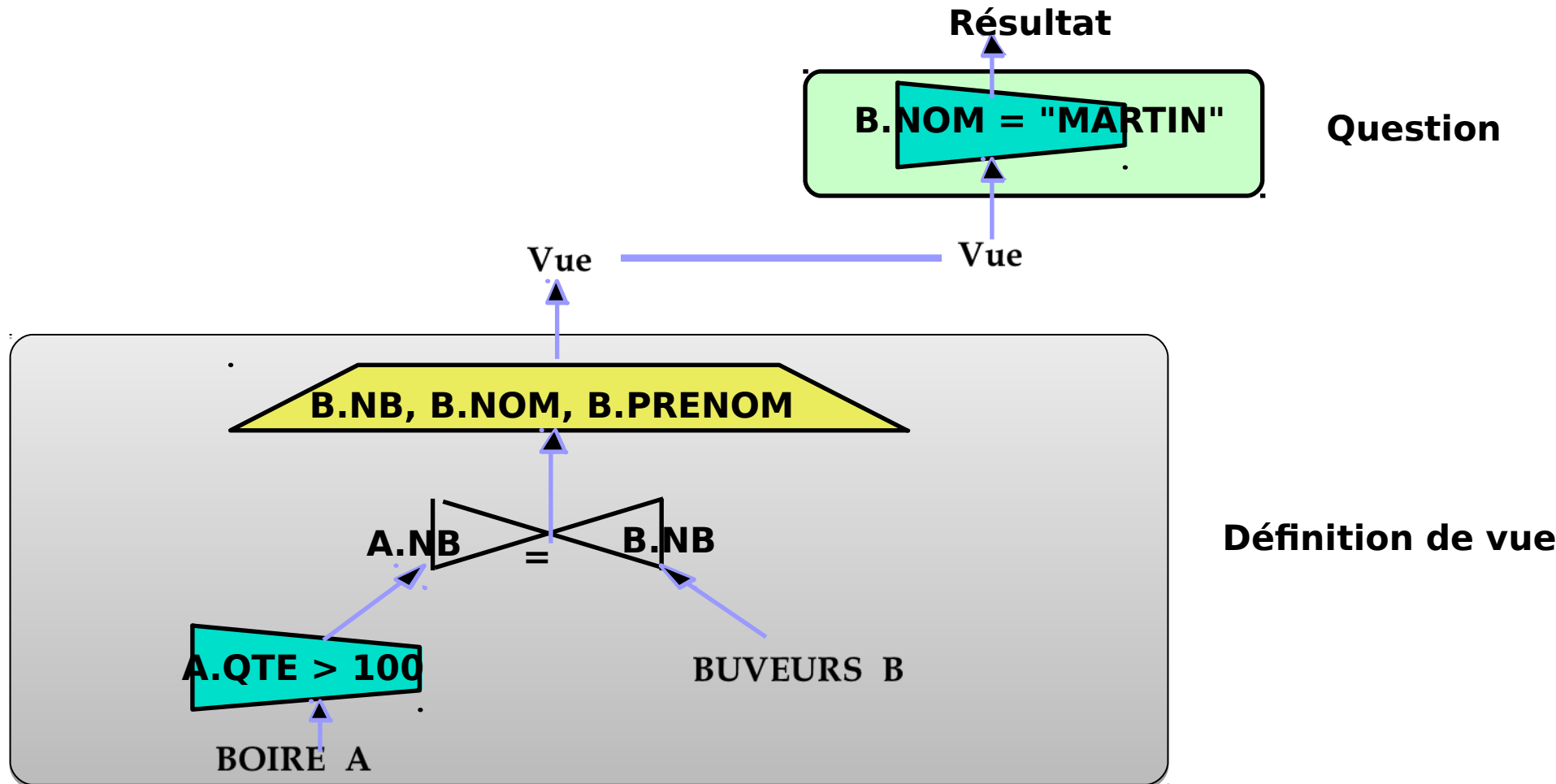
- La clause "WITH CHECK OPTION" demande la vérification du critère lors des insertions et mises à jour via la vue

- **CREATE VIEW GROS-BUVEURS (NB, NOM, PRENOM)**

**AS SELECT NB,
NOM, PRENOM
FROM BUVEURS B,
ABUS A**

**WHERE B.NB = A.NB
AND A.QTE > 100**

Modification de questions

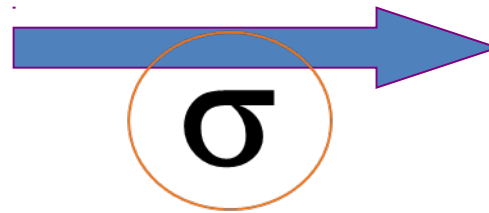


4. Exécution et Optimisation

- **Traduction automatique des questions déclaratives en programmes procéduraux :**
 - ➔ Utilisation de l'algèbre relationnelle
- **Optimisation automatique des questions**
 - ➔ Utilisation de l'aspect déclaratif de SQL
 - ➔ Gestion centralisée des chemins d'accès (index, hachages, ...)
 - ➔ Techniques d'optimisation poussées
- **Economie de l'astuce des programmeurs**
 - milliers d'heures d'écriture et de maintenance de logiciels.

Sélection

Patients			
Id-P	Nom	Prénom	Ville
1	Kamga	Jacques	Buea
2	Bouba	Zoe	Obala
3	Enow	John	Buea
4	Nlend	Paule	Loum

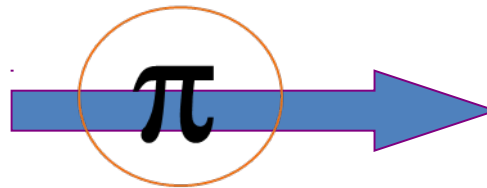


Patients			
Id-P	Nom	Prénom	Ville
1	Kamga	Jacques	Buea
2	Bouba	Zoe	Obala
3	Enow	John	Buea
4	Nlend	Paule	Loum

Patients de la ville de Buea

Projection

Patients			
Id-P	Nom	Prénom	Ville
1	Kamg a	Jacque s	Buea
2	Boub a	Zoe	Obala
3	Enow	John	Buea
4	Nlend	Paule	Loum



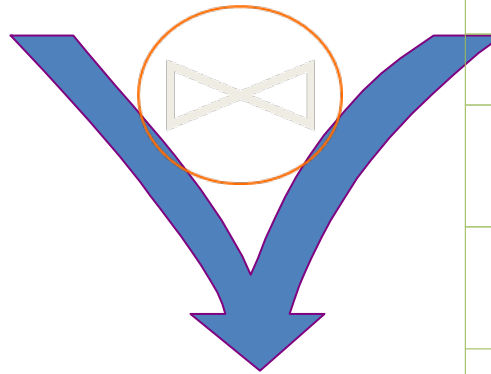
Patients			
Id-P	Nom	Préno m	Ville
1	Kamg a	Jacque s	Buea
2	Boub a	Zoe	Obala
3	Enow	John	Buea
4	Nlend	Paule	Loum

Nom et prénom des patients

Jointure

Patients			
Id-P	Nom	Prénom	Ville
1	Kamga	Jacques	Buea
2	Bouba	Zoe	Obala
3	Enow	John	Buea
4	Nlend	Paule	Loum

Visites				
Id-D	Id-P	Id-V	Date	Prix
1	2	1	15 juin	250
1	1	2	12 août	180
2	2	3	13 juillet	350
2	3	4	1 mars	250

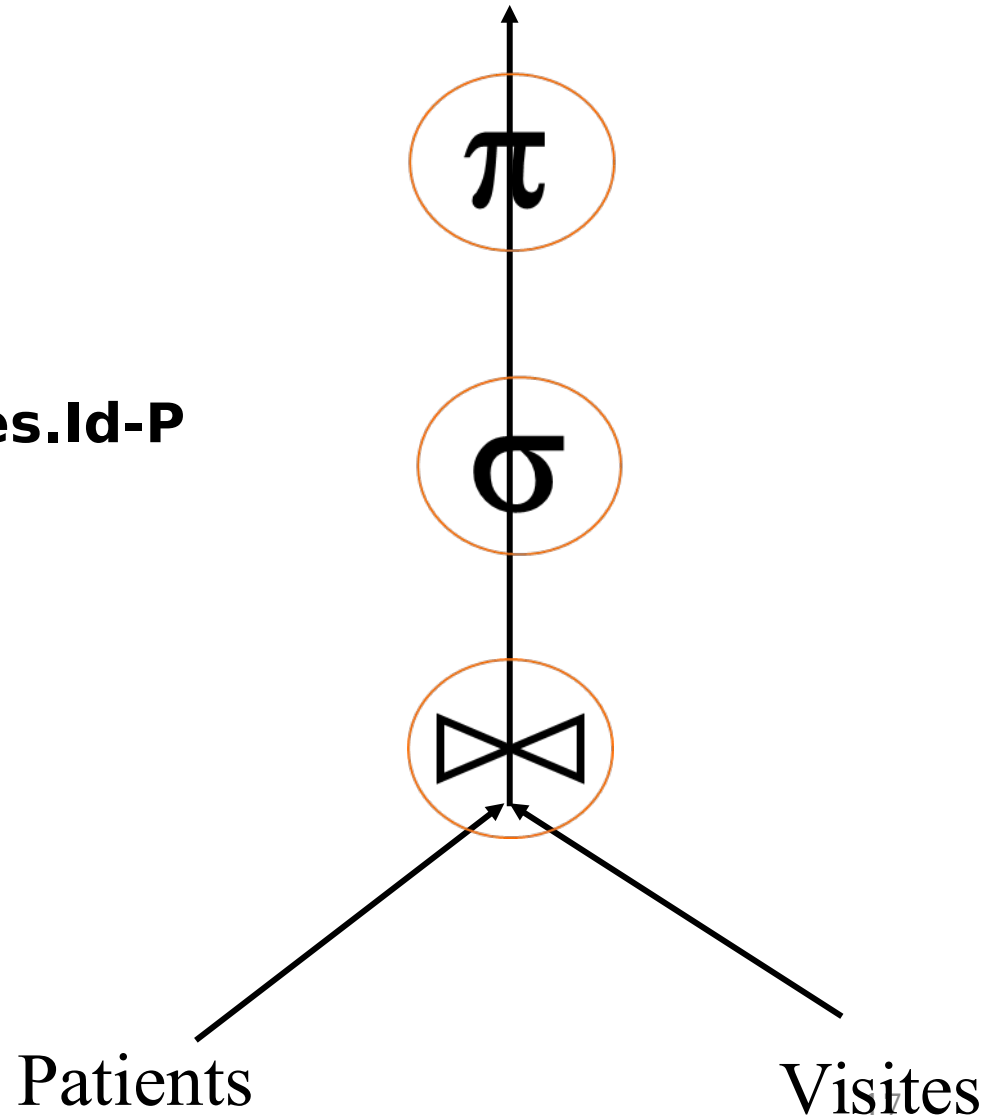


Id-P	Nom	Prénom	Ville	Id-D	Id-P	Id-V	Date	Prix
1	Kamga	Jacques	Buea	1	1	2	12 août	180
2	Bouba	Zoe	Obala	1	2	1	15 juin	250
2	Bouba	Zoe	Obala	2	2	3	13 juillet	350
3	Enow	John	Buea	2	3	4	1 mars	250

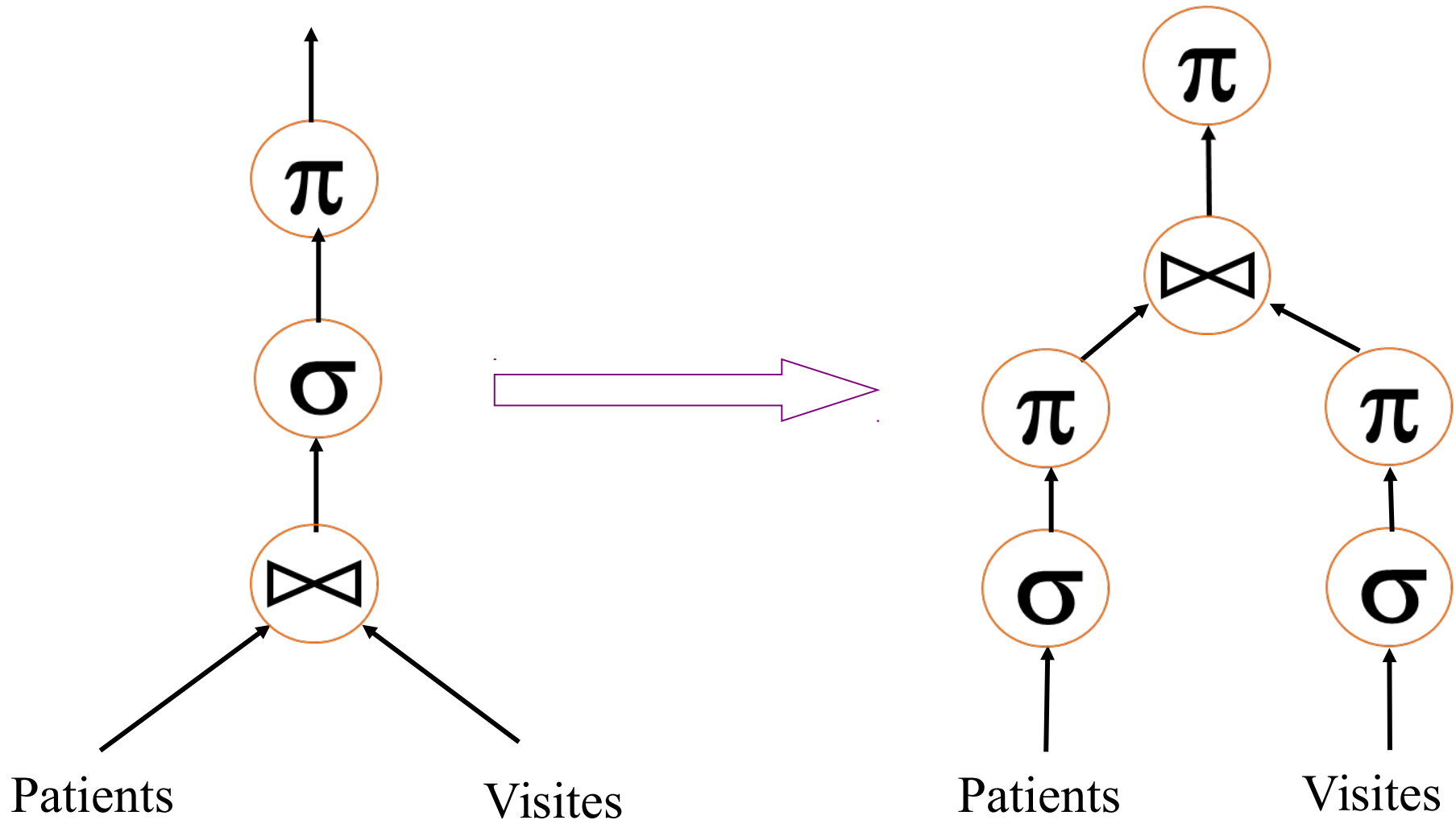
Patients et leurs visites

Exemple de plan d'exécution

Select Patients.Nom,
Patients.Prénom
From Patients, Visites
Where Patients.Id-P = Visites.Id-P
and Patients.Ville = 'Buea'
and Visites.Date = '15 juin'



Plan d'exécution optimisé



5. Intégrité Logique

- **Objectif : Détecter les mises à jour erronées**
- **Contrôle sur les données élémentaires**
 - Contrôle de types: ex: Nom alphabétique
 - Contrôle de valeurs: ex: Salaire mensuel entre 5 et 50kf
- **Contrôle sur les relations entre les données**
 - Relations entre données élémentaires:
 - Prix de vente > Prix d'achat
 - Relations entre objets:
 - Un électeur doit être inscrit sur une seule liste électorale

Contraintes d'intégrité

- **Avantages :**

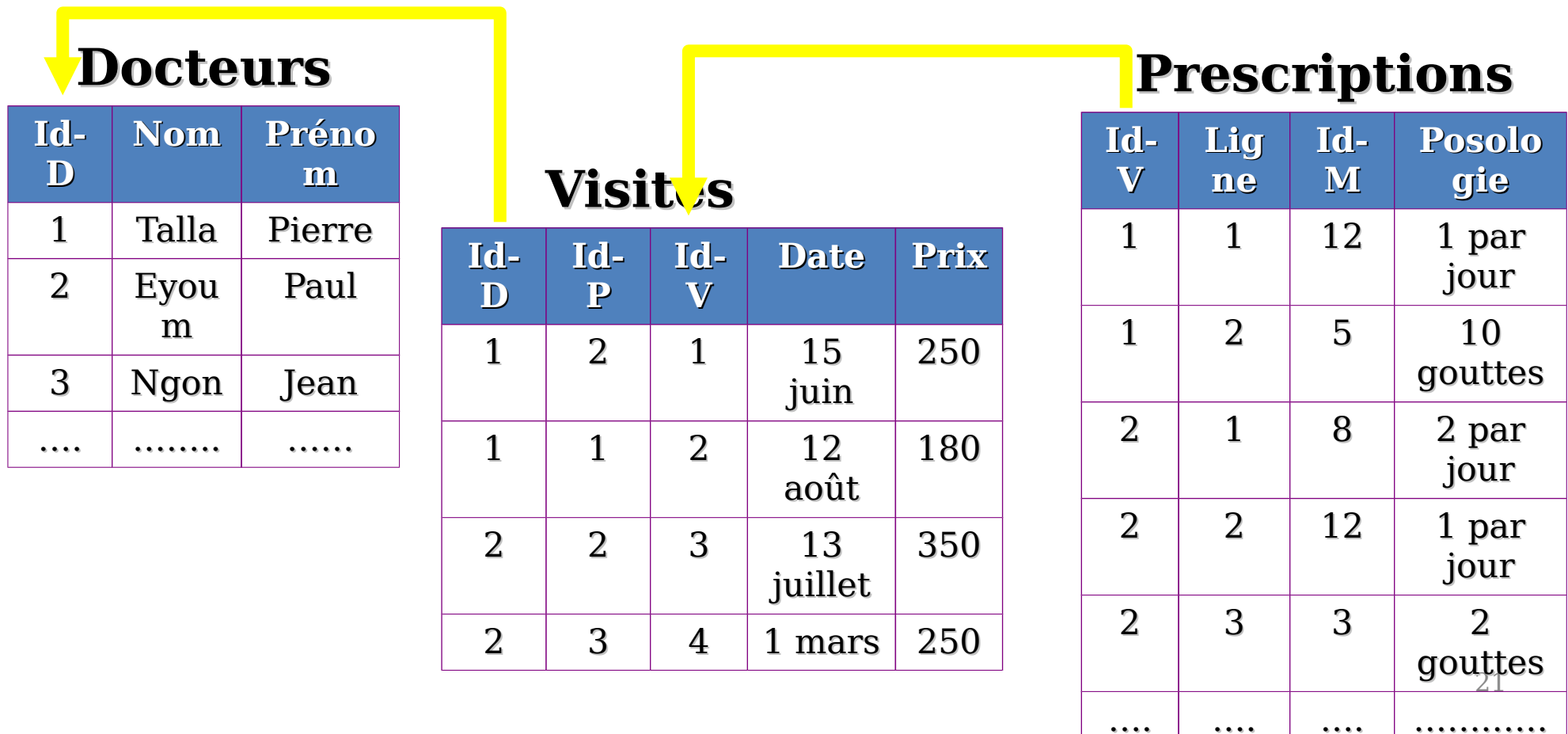
- **simplification** du code des applications
- **sécurité renforcée** par l'automatisation
- **mise en commun** des contraintes

- **Nécessite :**

- un langage de définition de contraintes d'intégrité
- la vérification **automatique** de ces contraintes

Exemples de contrainte

- Contraintes d'intégrité référentielles



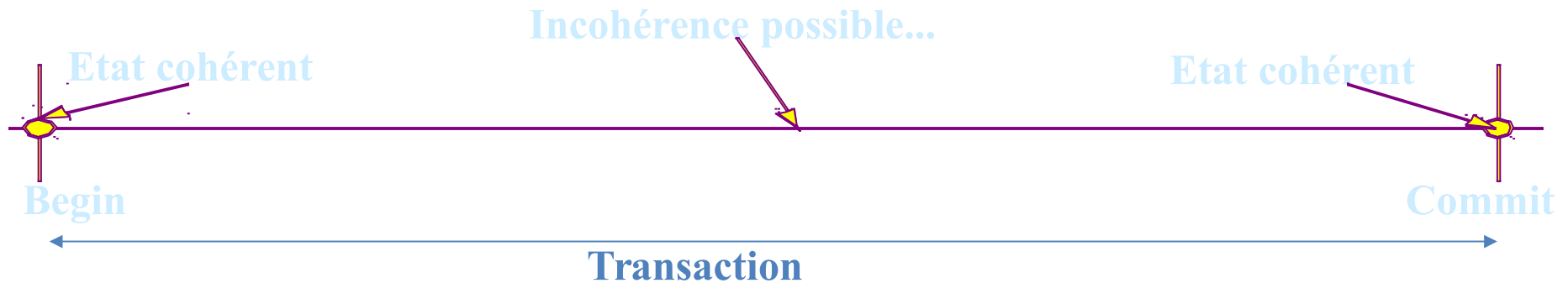
Techniques de contrôle

- **domaine de valeurs**
 - vérification à la volée
- **unicité de clé**
 - test de non existence dans l'index
- **références inter-tables**
 - insertion référençante : test d'existence dans l'index
 - suppression référencée: test de non existence (index ?)
- **assertion logique (quantificateurs, agrégats)**
 - très complexe
 - intérêt de gérer des agrégats pré-calculés

6. Intégrité Physique

- **Motivations : Tolérance aux fautes**
 - Transaction Failure : Contraintes d'intégrité, Annulation
 - System Failure : Panne de courant, Crash serveur ...
 - Media Failure : Perte du disque
 - Communication Failure : Défaillance du réseau
- **Objectifs :**
 - Assurer l'**atomicité** des transactions
 - Garantir la **durabilité** des effets des transactions commises
- **Moyens :**
 - Journalisation : Mémorisation des **états successifs** des données
 - Mécanismes de reprise

Transaction



Begin

$CE_{pargne} = CE_{pargne} - 3000$

$CC_{courant} = CC_{courant} + 3000$

Commit T1

Atomicité et Durabilité

ATOMICITE

Begin

CEpargne = CEpargne -
3000

CCourant = CCourant +
3000

Commit T1

Panne



→ Annuler le débit !!

DURABILITE

Begin

CEpargne = CEpargne -
3000

CCourant = CCourant +
3000

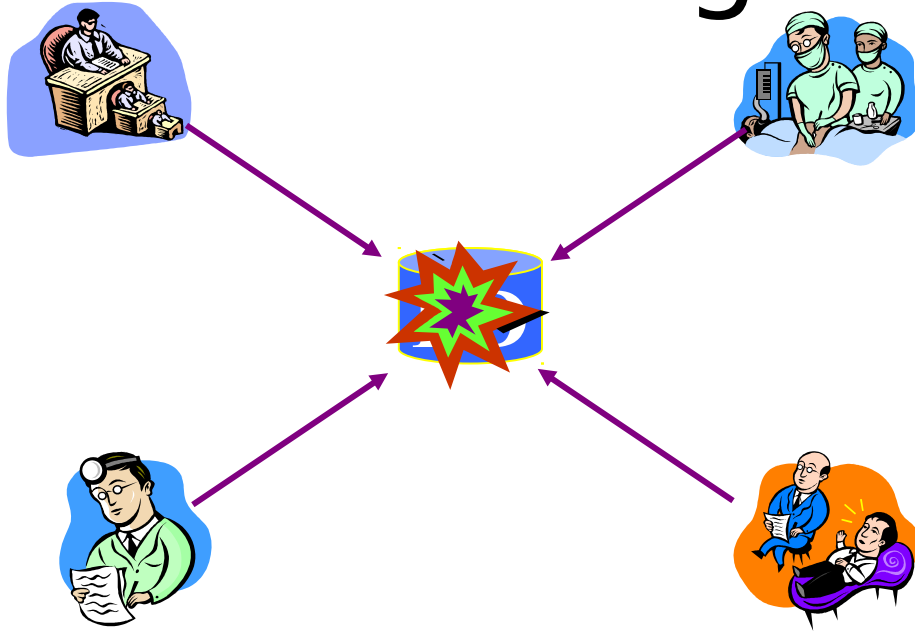
Commit T1

Crash disque



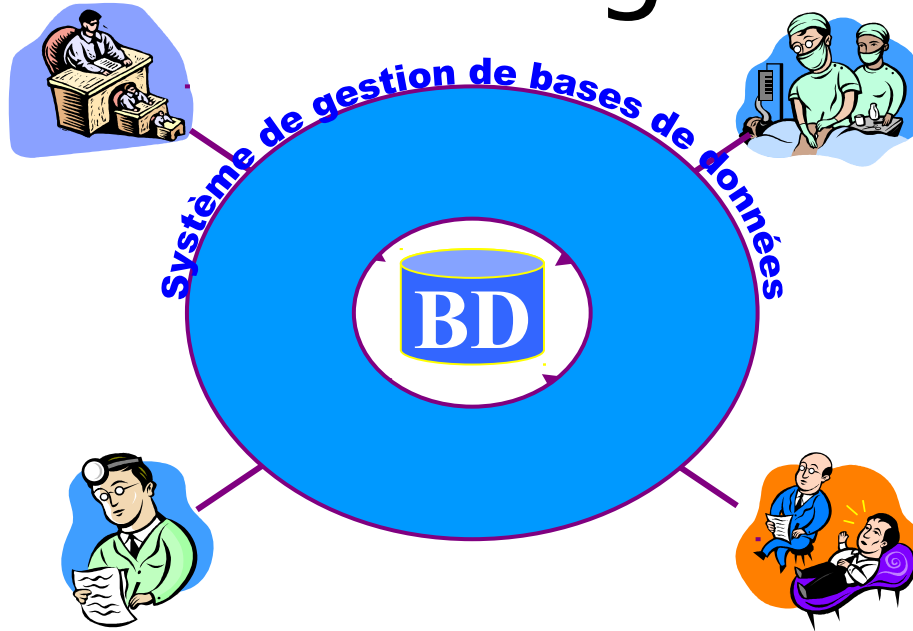
→ S'assurer que le
virement a été fait !

7. Partage des données



- Accès concurrent aux mêmes données
➔ Conflits d'accès !!

Partage des données



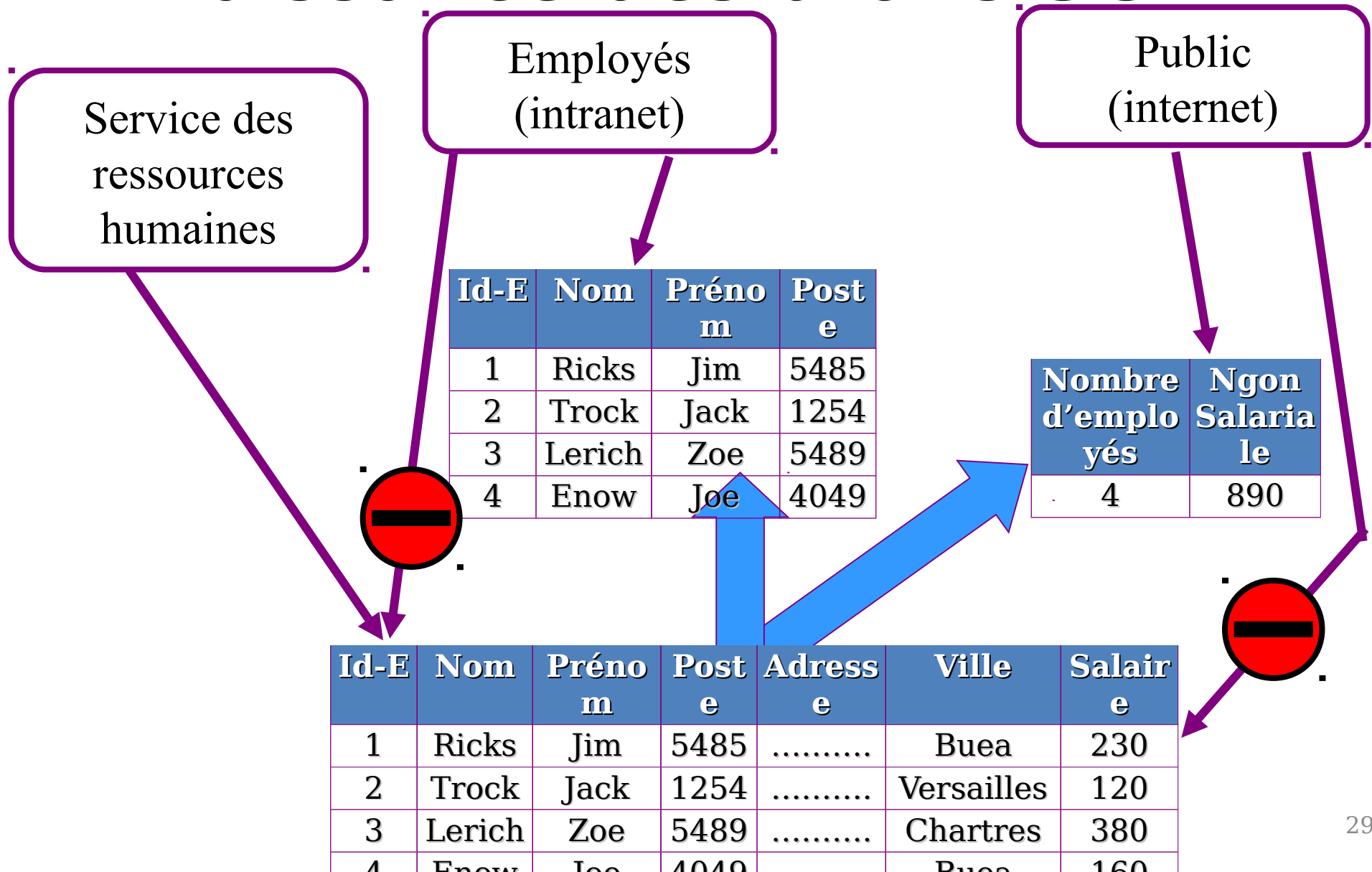
- **Technique de base**

- Le SGBD verrouille les données accédées pour l'utilisateur
- Il relâche les verrous en fin de transaction
- Distinction verrou en lecture et en écriture
- - Compatibilité lecture-lecture

8. Confidentialité

- **Objectif : Protéger les données de la BD contre des accès non autorisés**
- **Deux niveaux :**
 - Connexion restreinte aux **usagers répertoriés** (mot de passe)
 - **Privilèges** d'accès aux objets de la base
- **Usagers : Usager ou groupe d'usagers**
- **Objets : Relation, **Vue**, autres objets (procédures, etc.)**

Puissance des droits SGBD



Commandes de SQL

- **Attribuer autorisation**

- GRANT <DROITS> ON <RELATION>
- TO (<SUJET>)
- [WITH GRANT OPTION]
- <DROITS > ::= ALL | [<OPERATION>]...
- <OPERATION> ::= SELECT| INSERT| DELETE | UPDATE
| ALTER

- **Retirer autorisations**

- REVOKE <DROITS> ON <RELATION>
- FROM (<SUJET>)

- **Remarque**

- REVOKE DOIT RETIRER AUSSI LES DROITS TRANSMIS

9. Principaux SGBD

- **Les SGBD mettent en œuvre des techniques similaires**
- **Aujourd'hui 3 leaders :**
 - Oracle, IBM, Microsoft
- **Développements vers le e-business**
 - Site Web dynamiques
 - Commerce électronique BtoC
 - Commerce électronique BtoB
 - Support des documents (XML) ...

Principaux SGBD

- **Les grands SGBD**

- Oracle
- IBM DB2
- Microsoft SQL Server
- Sybase SQL Server
- Ingres
- Informix

- **Les open sources**

- MySQL
- PostgreSQL

- **Les SGBD personnels**

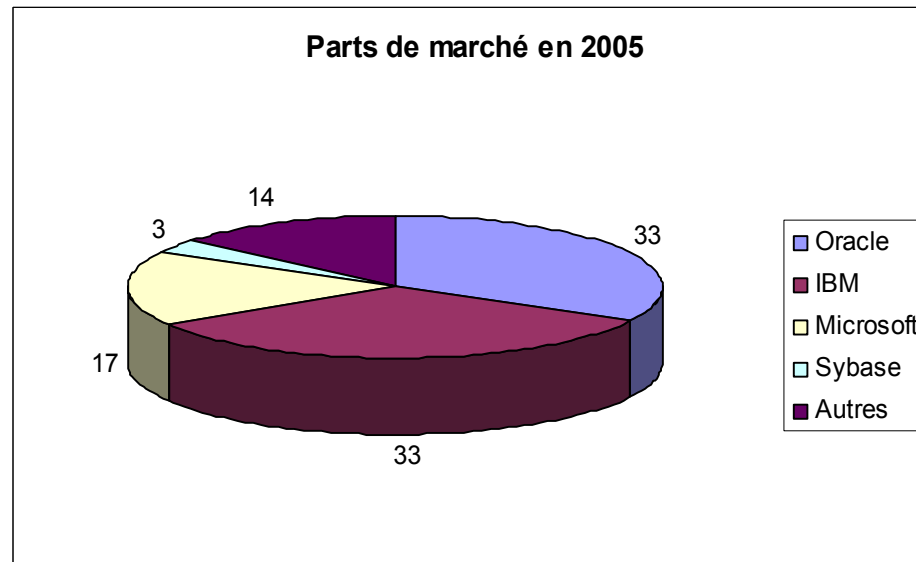
- Borland Paradox
- Filemaker
- Interbase
- Microsoft Access
- Microsoft FoxPro

- **Les SGBD objets**

- Objectivity
- Object Store
- Versant
- O2

Le marché des SGBD

- **Parts de marché**



Source: Dataquest