

# Base de données réparties

Pr. EL KABTANE Hamada

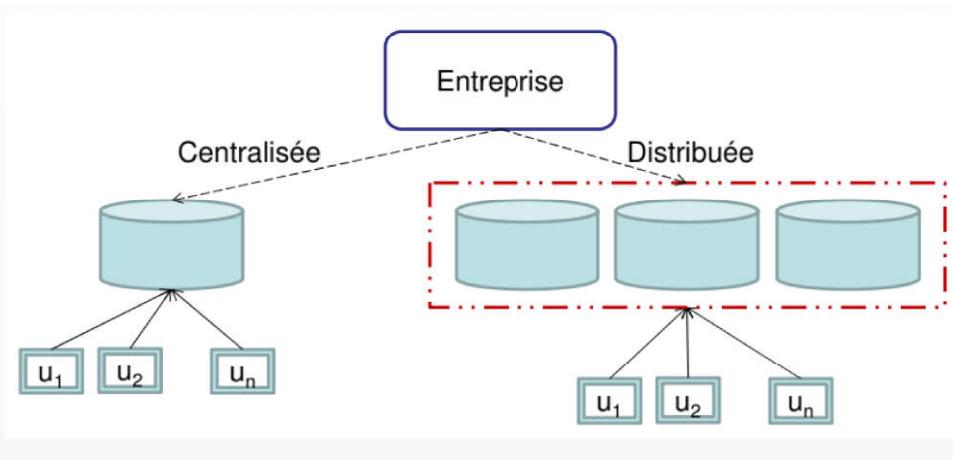


## Plan du cours

- ❑ Définition d'une BDR
- ❑ Principe fondamental d'une BDR
- ❑ Architecture des schémas
- ❑ Conception de BDR
- ❑ Fragmentation répartie
- ❑ Duplication répartie (Synchrone et asynchrone)
- ❑ Transactions réparties
- ❑ Contrôle de concurrence réparti
- ❑ Optimisation de requête répartie

# Introduction

Une entreprise, selon son besoin, peut choisir entre l'utilisation de base de données centralisée ou distribuée.



3

# Introduction

## Quelques définitions

- ▶ Une base de données distribuée est un ensemble de bases de données localisées et gérées par des sites différents et apparaissant à l'utilisateur comme une base unique.
- ▶ Une base de données distribuée est une collection de multiple bases de données distribuées sur un réseau informatique logiquement inter-reliées.
- ▶ Un client d'un système de gestion de base de données distribuée est une application qui accède aux informations distribuées par les interfaces du système.

4

# Introduction

## Quelques définitions (suite)

- Un système de gestion de base de données distribuées est un système qui gère des collections de BD logiquement reliées, distribuées sur un réseau, en fournissant un mécanisme d'accès qui rend la répartition transparente aux utilisateurs.
- Un système de gestion de bases de données distribuées (Distributed DBMS) est l'application qui permet la gestion de la base de données distribuée et rend la distribution transparente aux utilisateurs.

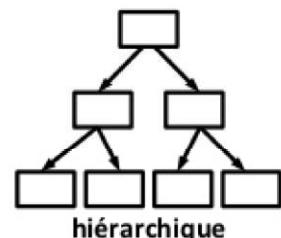
Pour référencer la BDD distribuée et le DBMS distribuées nous utilisant l'abréviation DDBS (Distributed DataBase System).

5

## Types des bases de données

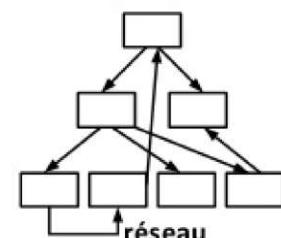
### Base hiérarchique

Lie les enregistrements dans une structure arborescente où chaque enregistrement n'a qu'un seul possesseur.



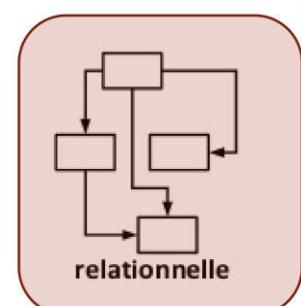
### Base en réseau

Est une base hiérarchique mais permet en plus d'établir des relations transverses.



### Base relationnelle

Stocke les informations décomposées et organisées dans des matrices appelées relations ou tables.



### Base objet

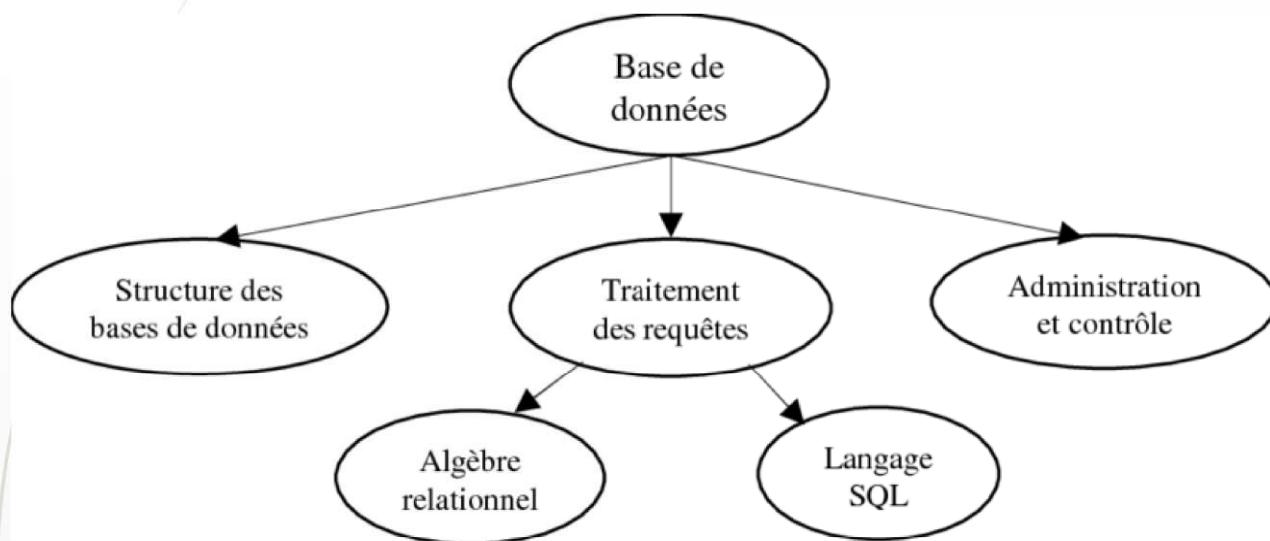
Stocke les informations groupées sous forme de collections d'objets persistants.

### Base XML

S'appuie sur le modèle de données fourni par XML.

6

# Bases de données relationnelles



7

## Bases de données réparties (BDR)

### ► Différents niveaux de répartition

- **Données**
- **Schémas** de la base de données
- **SGBD**
- **Traitement (requêtes, transactions)**
- **Composants matériels:** mémoires, disques, ...

# Terminologies

## ► BD répartie (**distributed database**) = BDR = BD + Réseau

- Ensemble de BDs gérées par des sites différents et qui apparaissent à l'utilisateur comme une base unique.

« *To the user, a distributed system should look exactly like a non distributed system* »

(Chris. Date, Introduction to Database Systems)

## ► SGBD Réparti (SGBDR)

- Logiciel qui gère une BDR et qui rend la répartition transparente

## ► Client de SGBDR

- Application qui accède aux informations distribuées par les interfaces du SGBDR

9

# Applications

## ► Cas de grosses entreprises ou organismes ayant des agences géographiquement distribuées:

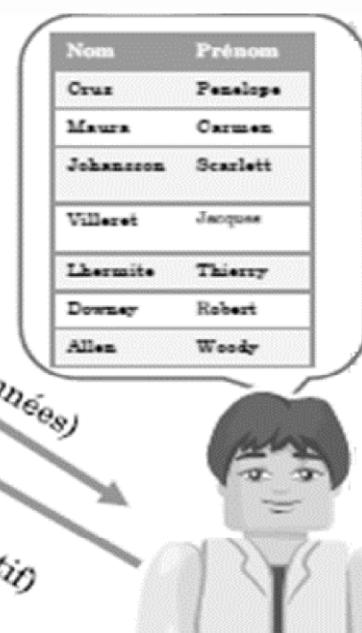
- Banques
- Fabrication
- Médicales (BD biologiques)
- Militaires
- Systèmes de réservation de compagnies aériennes
- WEB

# JUSQU'À PRÉSENT ...

Quels sont les noms et prénoms de tous les acteurs?

**BD Relationnelle**

Acteur		
Nom	Prénom	Nationalité
Cruz	Penelope	ESP
Maury	Carmen	ESP
Johansson	Scarlett	USA
Villeret	Jacques	FR
Lhermite	Thierry	FR
Downey	Robert	UK
Allen	Woody	USA



11

# ET MAINTENANT ?!?

**Londres**

Nom	Prénom	Nationalité
Downey	Robert	UK

**New York**

Nom	Prénom	Nationalité
Johansson	Scarlett	USA
Allen	Woody	USA

**Madrid**

Nom	Prénom	Nationalité
Cruz	Penelope	ESP
Maury	Carmen	ESP

**Paris**

Nom	Prénom	Nationalité
Villeret	Jacques	FR
Lhermite	Thierry	FR

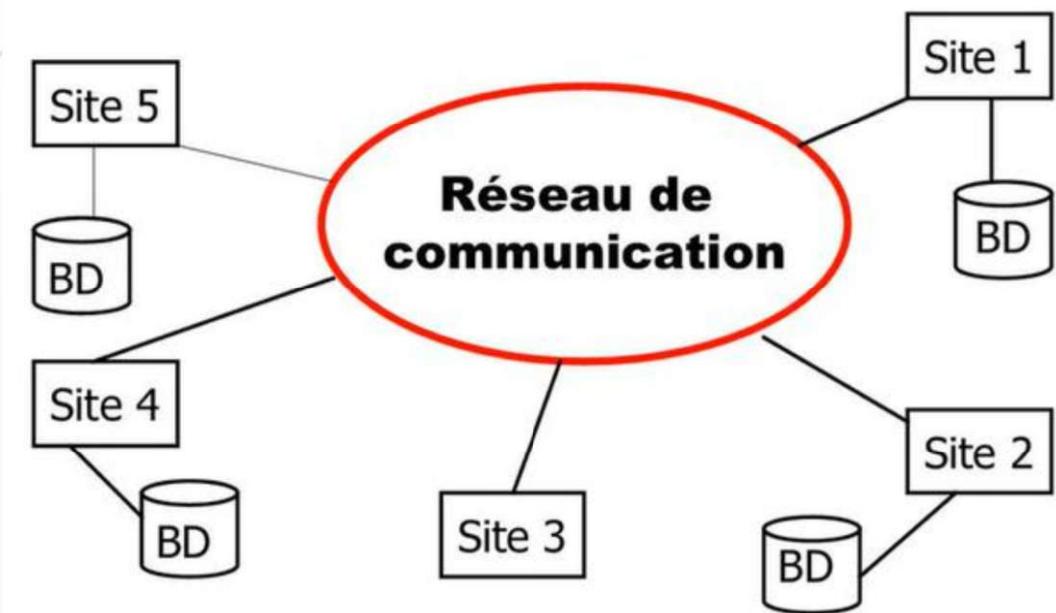


Quels sont les noms et prénoms de tous les acteurs?

12

# DÉFINITION DU CONTEXTE

- Un ensemble de données stockées sur plusieurs sites (machines) communiquant via un réseau



13

## Problématique et Objectifs

- **Exemple:**

- Soit une entreprise qui a des annexes dans plusieurs villes: Marrakech, Casablanca, Agadir Le gérant veut maintenir une base de données des employés et les projets sur lesquels ils travaillent.

Le gérant utilise 4 tables:

- Emp (idEmp, nomEmp, poste);
  - Prj (idPrj, nomPrj, designation, budget, ville);
  - Sal (poste, salaire);
  - Aff (idEmp, idPrj, resp, duree) //Affectation d'un employé à un projet avec telle responsabilité et telle durée.

Si la base de données est centralisée, on peut trouver les noms des employés qui travaillent dans un projet pour une durée supérieure de 12 mois ainsi que leurs salaires avec la requête suivante:

```
SELECT nomEmp, salaire  
FROM Emp, Aff, Sal  
WHERE Aff.duree > 12  
AND Emp.idEmp = Aff.idEmp  
AND Sal.poste = Emp.poste
```

14

# Problématique et Objectifs

- **Exemple (suite):**

Maintenant, si la base est distribuée sur les différents annexes, de telle sorte que les informations des employés et les projets de chaque annexe sont stockés dans une base de données localisée dans cette annexe. L'utilisateur doit poser la même question au système sans faire attention à la distribution des informations (transparence), et c'est le système qui s'occupe de la récupération des informations.

- **Problématique:**

- Comment offrir aux utilisateurs un accès à l'ensemble des données en **faisant abstraction de leur localisation physique** tout en garantissant un traitement efficace des requêtes et le respect de contraintes (physiques) géographiques ?

15

# Problématique et Objectifs

- **Objectifs:**

- Autonomie locale
- Transparence
- Performance améliorée
- Fiabilité et disponibilité accrues
- Partage accru de données et ressources
- Expansion graduelle

16

# Problématique et Objectifs

- Pour cela, il a été défini plusieurs types de transparence:

## 1/ Transparence de données (Indépendance)

La définition des données dans une base de données se fait à deux niveaux (structure logique et structure physique), ce qui permet de définir une indépendance de données logique et indépendance de données physique.

L'indépendance de données logique signifie que les applications de l'utilisateur ne sont pas autorisées à modifier la structure logique des données (le schéma).

L'indépendance de données physique signifie que les détails de la structure de stockage des données seront cachés aux applications de l'utilisateur.

17

# Problématique et Objectifs

## 2/ Transparence du réseau (ou de distribution)

Ce type de transparence fait référence à ce que l'existence du réseau est caché à l'utilisateur. L'accès aux données de la base doit se faire de la même manière soit en mode centralisé ou distribué.

18

# Problématique et Objectifs

## 3/ Transparence de réPLICATION (redondance)

Pour des raisons de performance et disponibilité des données, il est préférable de faire des copies de quelques données de la base sur plusieurs sites.

Par exemple, les données qui se trouvent sur un autre site et qui sont réclamées souvent par un utilisateur, sera préférable d'en faire une copie sur la base de données local de cet utilisateur.

En plus, si une machine s'arrête, on peut trouver les données sur une autre machine.

La décision de réPLICATION des données, et de combien de copies doit-on faire, dépend des applications des utilisateurs. L'utilisateur n'a pas besoin de savoir si les données sont répliquées ou non. Il envoie ses requêtes comme s'il existe une seule copie des données.

19

# Problématique et Objectifs

## 3/ Transparence de fragmentation

Dans certains cas, et pour des raisons de performance, on aura besoin de fragmenter une table de la base de données et placer chaque fragment sur un site différent.

La fragmentation permet aussi de réduire les inconvénients de la réPLICATION.

Cette opération de fragmentation doit être faite d'une manière transparente à l'utilisateur. Il utilise la table originale comme si elle n'est pas fragmentée.

Il existe deux types de fragmentation:

- Fragmentation horizontale: La table est partitionnée en plusieurs sous-tables contenant chacune un sous-ensemble de lignes de la table originale.
- Fragmentation verticale: La table est partitionnée en plusieurs sous-tables qui sont définies sur un sous-ensemble d'attributs (de colonnes) de la table originale.

20

# Défis des BDR

- **Conception d'une BDR:**
  - Fragmentation
  - Allocation
  - RéPLICATION (totale ou partielle)
- **Transparence à la répartition:**
  - Extension de la notion d'indépendance logique et physique des données.
  - Location (réPLICATION, fragmentation): Aucune spécification de la localisation des données.

21

# Défis des BDR

- **Optimisation de requêtes réparties:**
  - Choix de la copie en lecture
  - Mise à jour de toutes les copies
  - Plan d'exécution répartie
- **Transaction réparties:**
  - Utilisateur aura à formuler ses transactions de la même manière que dans un environnement centralisé.
  - Maintien des propriétés ACID (atomicité, cohérence, isolation et durabilité) des transactions.

22



# ACID

- **Atomicité:**

- La propriété d'atomicité assure qu'une transaction se fait au complet ou pas du tout : si une partie d'une transaction ne peut être faite, il faut effacer toute trace de la transaction et remettre les données dans l'état où elles étaient avant la transaction.

- **Cohérence:**

- La propriété de cohérence assure que chaque transaction amènera le système d'un état valide à un autre état valide.

23



# ACID

- **Isolation:**

- Toute transaction doit s'exécuter comme si elle était la seule sur le système.

- **Durabilité:**

- La propriété de durabilité assure que lorsqu'une transaction a été confirmée, elle demeure enregistrée même à la suite d'une panne d'électricité, d'une panne de l'ordinateur ou d'un autre problème.

24

# Avantages de la distribution

- ▶ **Reflète une structure organisationnelle:** nombre d'organisations sont aujourd'hui réparties sur plusieurs sites.
- ▶ **Amélioration du partage :** distribution des données sur les différents sites; un utilisateur peut accéder aux données de son site tout en pouvant voir les données sur les autres sites.
- ▶ **Disponibilité et fiabilité améliorée:** une panne de site ne bloque pas tout le système. Les données dupliquées impliquent la possibilité d'accès en cas de panne d'un site.
- ▶ **Performances améliorées:** La parallélisme inhérent à un SGBD distribué implique une vitesse d'accès aux données meilleure que dans une BD centralisée.

25

# Avantages de la distribution

- ▶ **Flexibilité:** en ajoutant des sites au réseau le système réparti peut étendre la BD et enrichir les traitements pour de nouvelles applications.
- ▶ **Autonomie:** la répartition permet aux différentes structures d'une même entreprise de conserver leurs spécificités (méthodes de travail, perceptions ...)
- ▶ **Economies:** La distribution est synonyme de petits équipements, donc la création de système informatique distribué est à coût inférieur si le même système était déployé sur un gros équipement.

26

# Inconvénients de la distribution

- ▶ Le manque d'expérience des équipes de développement et des utilisateurs finaux.
- ▶ Coûts de déploiement (de communication, d'adaptation des applications et des hommes).
- ▶ Conception et développement complexe : fragmentation, allocation des fragments, réPLICATION etc.
- ▶ Contrôle d'intégrité plus difficile
- ▶ Sécurité: données dupliquées, réseaux

27

## Types de BDR

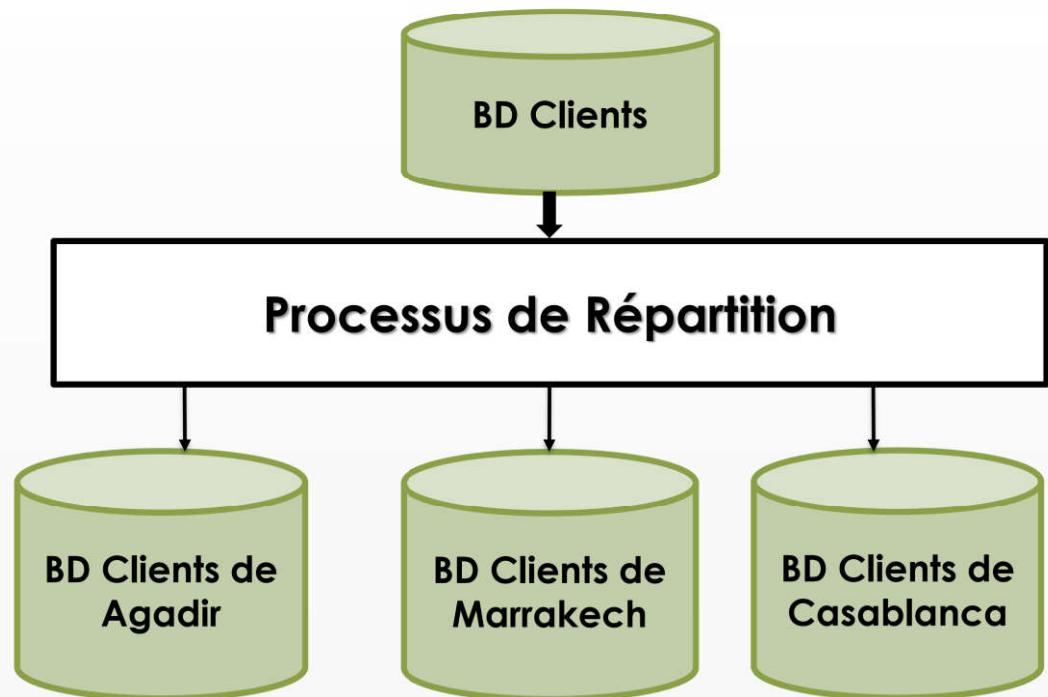
### □ BDR Homogène

- Obtenue en **divisant** une BD en un ensemble de BD locales, chacune étant gérée par le même SGBD
- Même modèle de données
- Même langage de requêtes
  - **Exemple:** DB2, ORACLE (SQL)
- Données de la base sont réparties sur plusieurs sites

28

## Types de BDR

### BDR Homogène



29

## Types de BDR

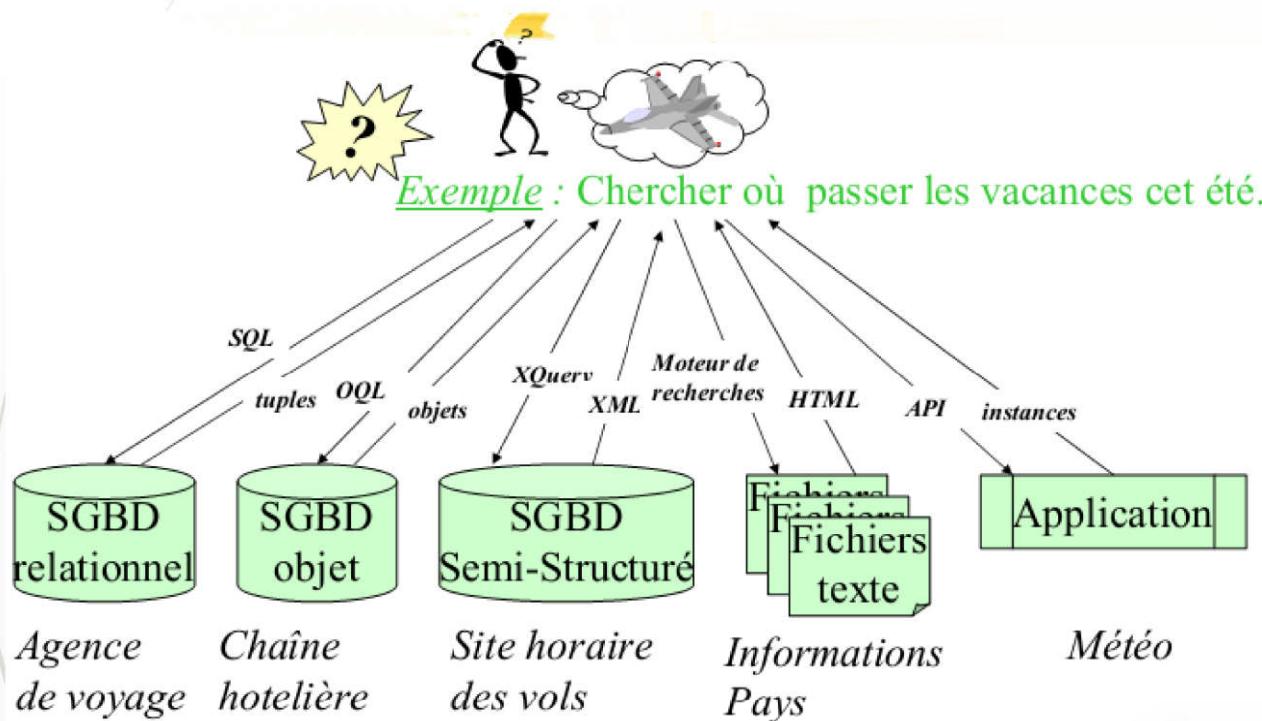
### BDR Hétérogènes

- Deux niveaux d'hétérogénéité:
  - Les BD ont le même modèle (relationnel) mais sont gérées par des **SGBD différents** (Oracle, SQL server, ....)
  - Les BD ont des **modèles différents** (relationnel, objet) et gérées par des **SGBD différents** (Oracle, O2)
- BDR hétérogène
  - BD répartie obtenue en **intégrant** dans une BD unique un ensemble de BD locales gérées par des SGBD différents.

30

# Types de BDR

## □ BDR Hétérogènes



31

## Pourquoi les BDR??

## □ Problème 1

- Une société d'assurance de santé est implantée sur l'ensemble du Maroc
- Le siège est localisé à Marrakech
- Dans un souci de service, la société crée de nouvelles filiales au Maroc
- Chaque filiale gère les dossiers des adhérents de sa région
- Chaque filiale peut accéder aux informations du siège ou d'une autre filiale.

32

# Pourquoi les BDR?

## Solution 1: Une seule BD centrale

### Avantages

- Un seul SGBD
- Un seul administrateur de la BD
- La requête est calculée sur le serveur par le SGBD et les résultats sont acheminés vers le client distant

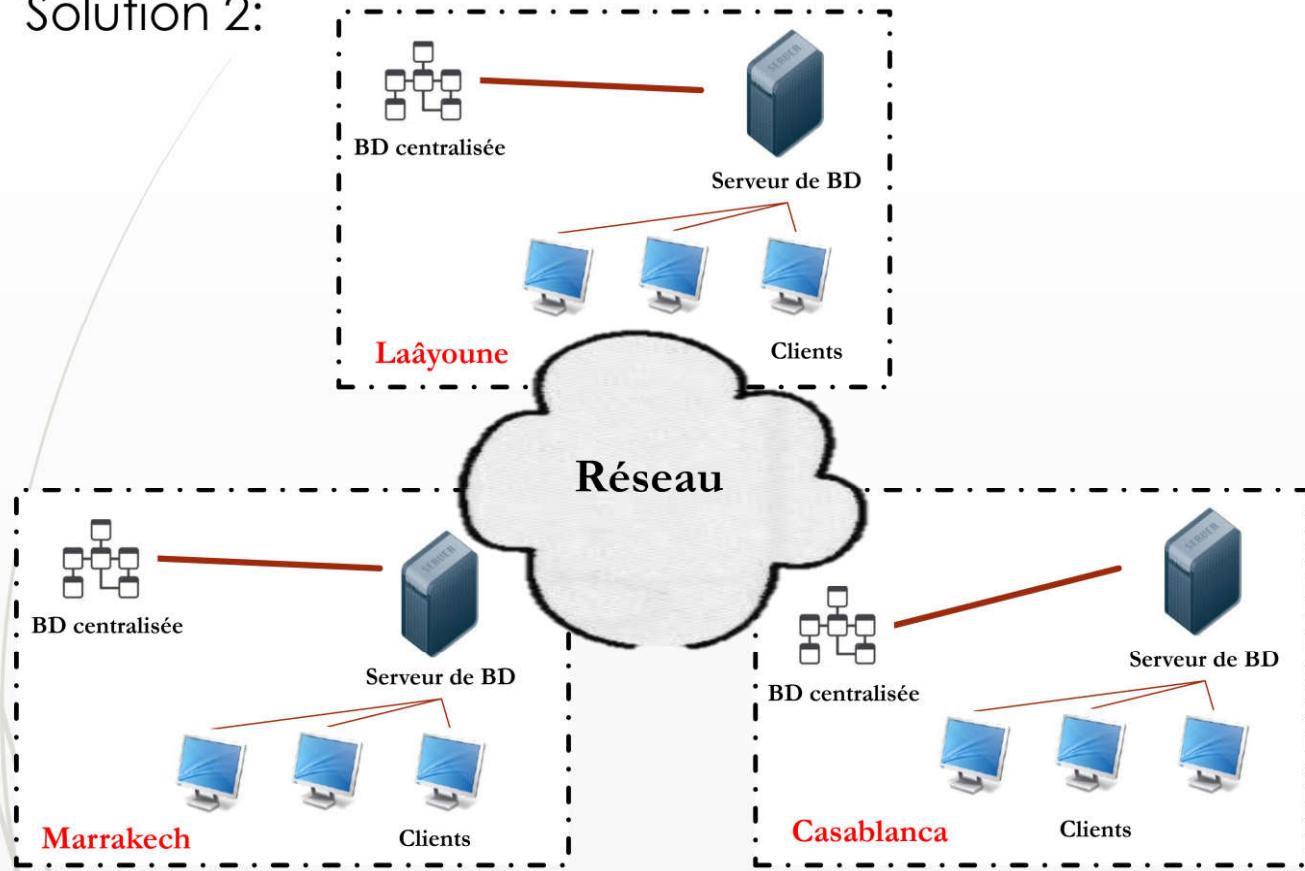
### Inconvénients

- Si le site central tombe en panne ?!
- Si le réseau tombe en panne ?!
- Le coût de la communication et du transfert des données
- Charge de calcul concentrée sur le serveur central

33

# Pourquoi les BDR?

## Solution 2:



34

# Pourquoi les BDR?

## Solution 2:

### Avantages

- Toute la base de données est localisée dans chaque site
- La requête est calculée sur le serveur local par le SGBD local
- Réduction du coût de la communication

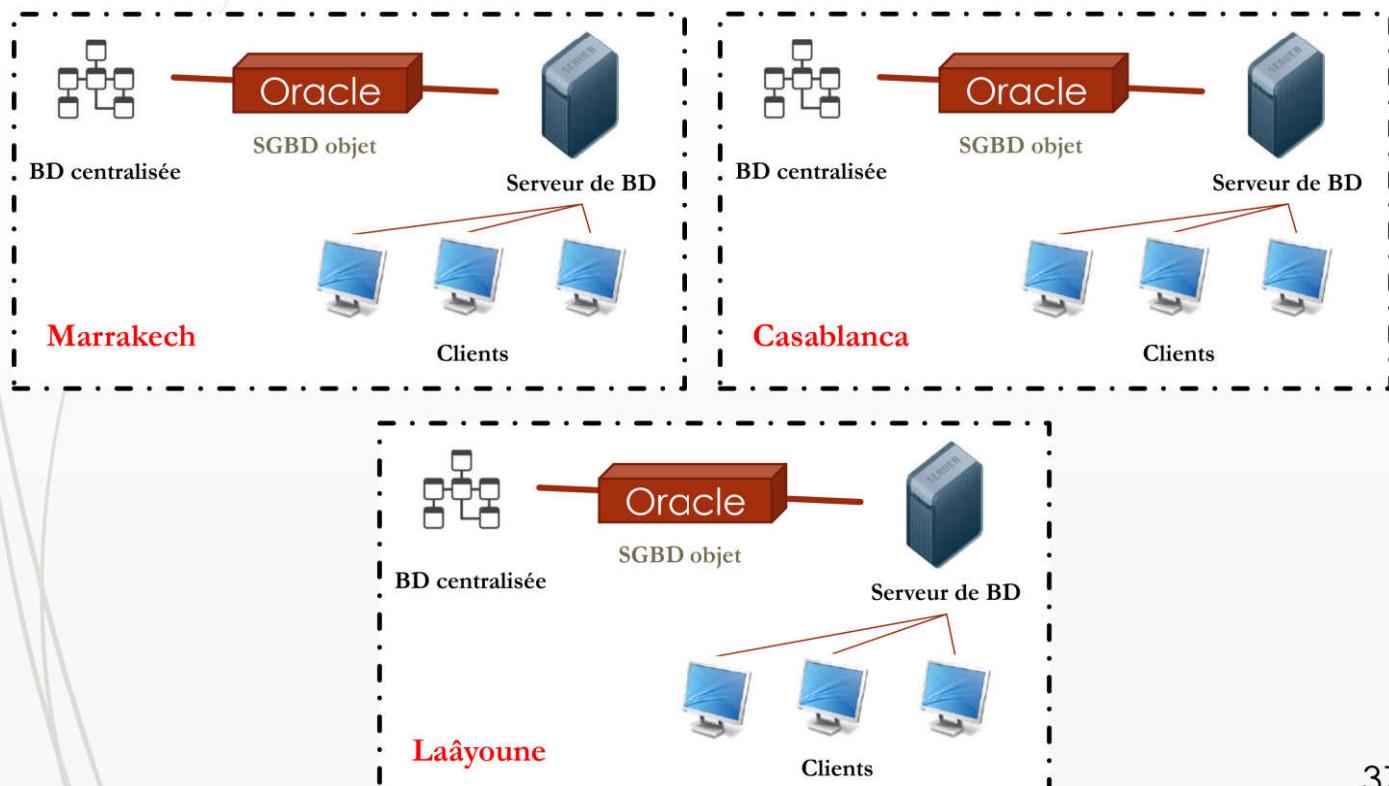
### Inconvénients

- Plusieurs SGBDs
  - ▶ Problèmes de MAJ et d'incohérence de la base
  - ▶ Plusieurs administrateurs de la base
- Surcharge de la base par des données non nécessairement utiles en local.

35

# Pourquoi les BDR?

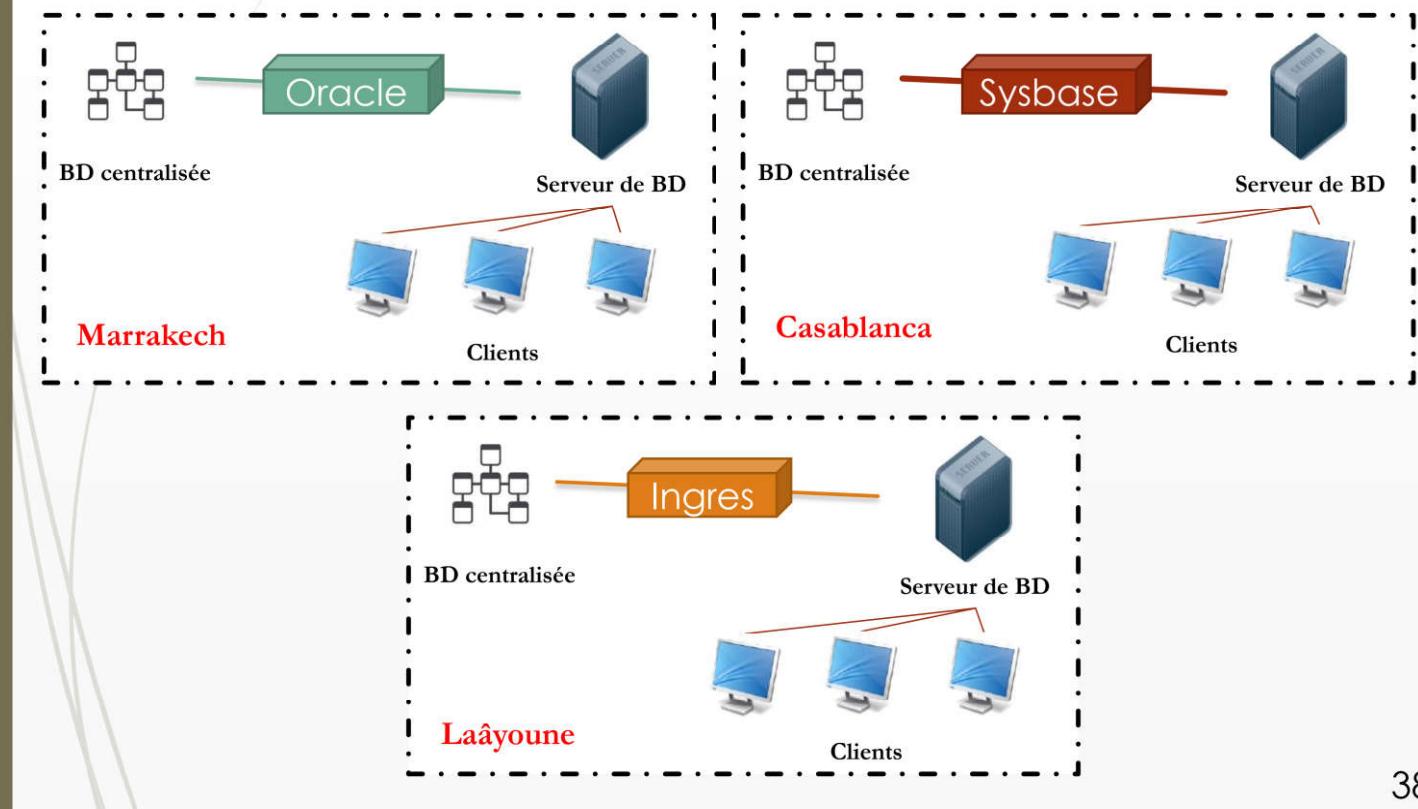
## □ PROBLEME 2.1: Distribution géographique



37

# Pourquoi les BDR?

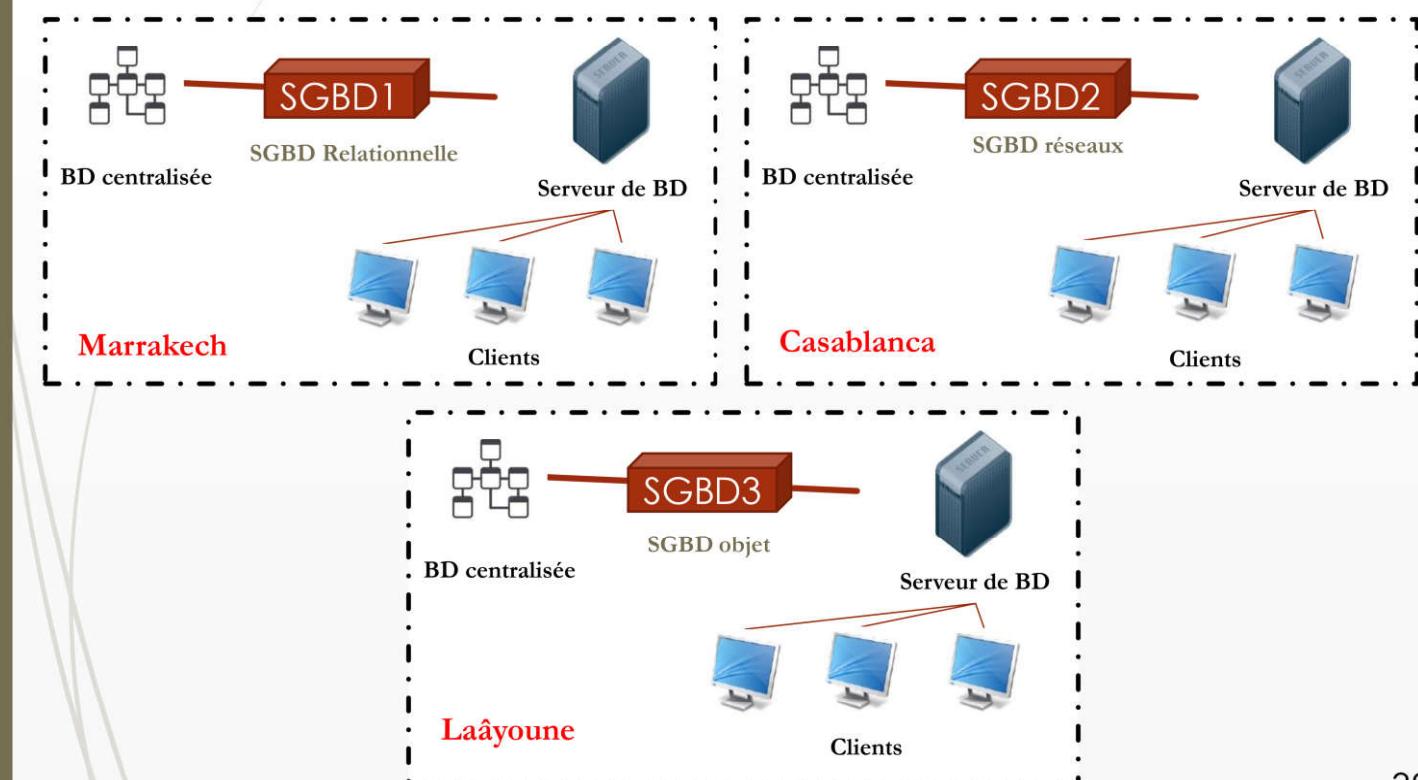
## □ PROBLEME 2.2: Hétérogénéité des SGBDS



38

# Pourquoi les BDR?

## □ PROBLEME 2.3: Hétérogénéité des modèles



39

## Principe fondamental d'une BDR

### Autonomie locale

La BD locale est complète et autonome (intégrité, sécurité, gestion), elle peut évoluer indépendamment des autres (upgrades...)

### Egalité entre sites

Un site en panne ne doit pas empêcher le fonctionnement des autres sites (mais perturbations possibles)

### Fonctionnement continu

Distribution permet résistance aux fautes et aux pannes (en théorie)

41

## Principe fondamental d'une BDR

### Localisation transparente

Accès uniforme aux données quel que soit leur site de stockage

### Fragmentation transparente

Des données (d'une même table) éparpillées doivent être vues comme un tout

### Indépendance à la réPLICATION

Les données répliquées doivent être maintenues en cohérence (délai possible)

42

## Principe fondamental d'une BDR

### Requêtes distribuées

L'exécution d'une requête peut être répartie (automatiquement) entre plusieurs sites (si les données sont réparties)

### Transactions réparties

Le mécanisme de transactions peut être réparti entre plusieurs sites (si ...)

### Indépendance vis-à-vis du matériel

Le SGBD fonctionne sur les différentes plateformes utilisées.

43

## Principe fondamental d'une BDR

### Indépendance vis-à-vis du SE

Le SGBD fonctionne sur les différents SE...)

### Indépendance vis-à-vis du réseau

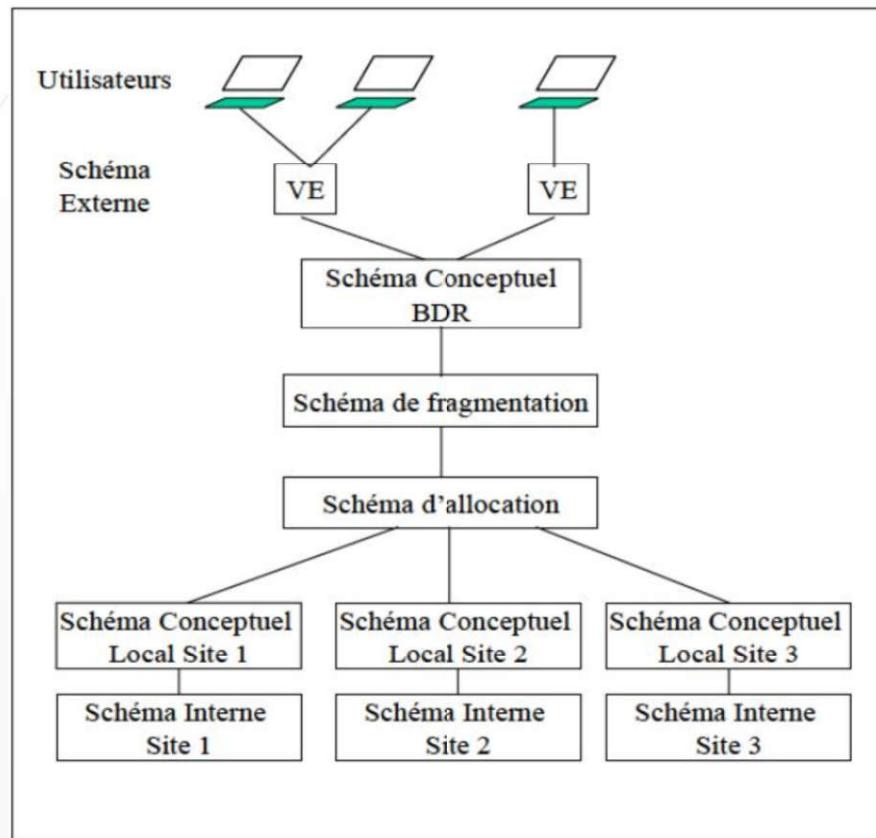
Le SGBD est accessible à travers les différents types de réseau utilisés

### Indépendance vis-à-vis du SGBD

La base peut être distribuée sur des SGBD hétérogènes

44

# Architecture d'une BDR



Architecture d'une BDR.

45

## Utilisation d'une BD répartie

- ▶ Le principe fondamental des BDR est **la transparence pour l'utilisateur**. Cette transparence s'exprime par **les formes suivantes** :
- ▶ **Les utilisateurs** accèdent à la BD
  - Soit directement par le **schéma conceptuel**
  - Soit indirectement au travers de **vues externes** (Figure précédente).
- ▶ Mais en aucun cas **les utilisateurs n'ont les moyens d'accéder aux schémas locaux ni de préciser le site**. C'est le principe de **transparence de localisation**.

46

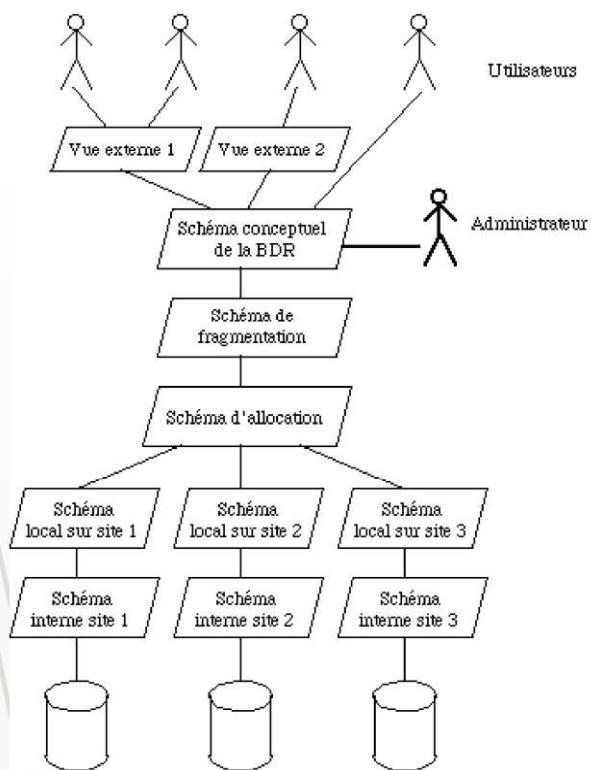
## Exemple

- Une banque peut posséder des agences à Marrakech et à Rabat.
  - **Dans une BD centralisée**, le siège social de la banque gérerait tous les comptes des clients et les agences devraient communiquer avec le siège social pour avoir accès aux données.
  - **Dans une BD répartie**, les informations sur les comptes sont distribuées dans les agences et celles-ci sont interconnectées (entièrement ou partiellement) afin qu'elles puissent avoir accès aux données externes .
- Cependant, **la répartition de la base de données bancaire est invisible** aux agences en tant qu'utilisateurs,



47

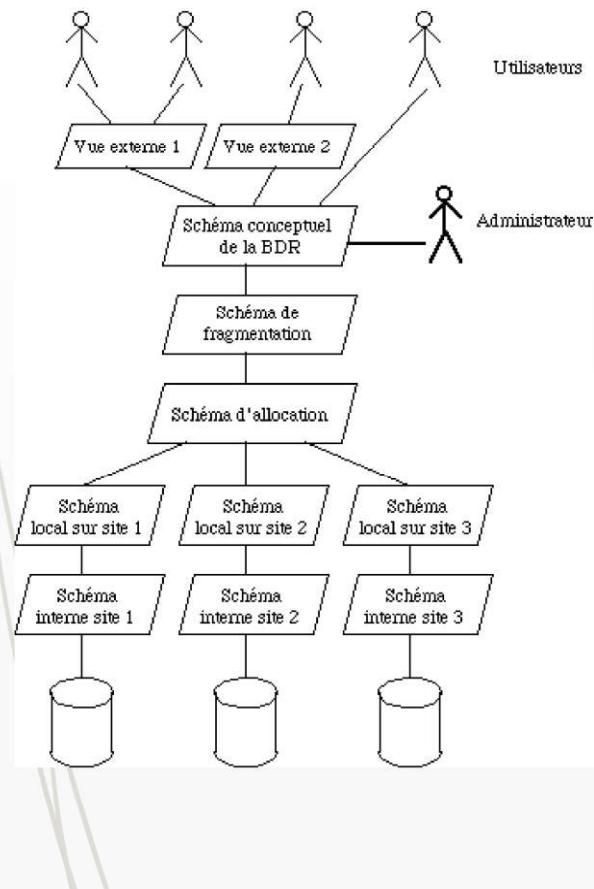
## Exemple



- **Transparence de partitionnement :** Un client peut ouvrir un compte à **Marrakech** et effectuer des opérations à **Casablanca**. C'est le **système qui recherche le site** où sont mémorisés ses informations et non l'utilisateur qui doit l'indiquer. En plus, **les utilisateurs n'ont pas à connaître les partitionnements** de la base de données. Les utilisateurs ne doivent pas se préoccuper de la **réunification des données fractionnées**. C'est le système qui gère les partitionnements et qui doit rechercher **toutes les partitions et les intégrer en une seule information logique** présentée à l'utilisateur.
- **Transparence de duplication des données:** Les utilisateurs n'ont pas à savoir si plusieurs copies d'une même information sont disponibles. Lors de la modification d'une information, c'est le système qui doit se préoccuper de mettre à jour toutes les copies

48

## Exemple



- Les utilisateurs accèdent aux données par l'intermédiaire du schéma conceptuel global et non par l'intermédiaire du schéma d'un site.
- C'est uniquement le système qui désigne le ou les sites qui sont concernés par la demande de l'utilisateur. L'indépendance physique, assurée dans les bases de données traditionnelles, est donc conservée dans les BD réparties.

49

## Architecture: Niveau de répartition

La répartition d'une base de données intervient dans les trois niveaux de son architecture en plus de la répartition physique des données :

- **Niveau externe** : les vues (schémas externes) sont distribuées aux utilisateurs sur leurs sites.
- **Niveau conceptuel** : le schéma conceptuel des données (schéma logique) est associé, par l'intermédiaire du schéma de répartition (lui même décomposé en un schéma de fragmentation et un schéma d'allocation), aux schémas locaux qui sont réparties sur plusieurs sites, les sites physiques.
- **Niveau interne** : le schéma interne global n'a pas d'existence réelle mais fait place à des schémas internes locaux répartis sur différents sites (en principe les sites d'accueil des schémas logiques répartis)

50

## Conception d'une BDR

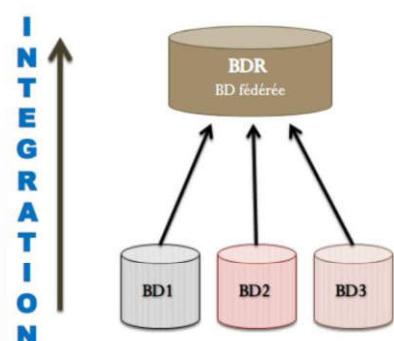
- Une base de données répartie est un système compliqué, qui s'il peut-être évité, il devrait l'être!
- Certaines circonstances dans la vie du SI nous impose la mise en place d'une base de donnée répartie.
- On distingue deux approches fondamentalement différentes
  - **Conception ascendante**
  - **Conception descendante**

51

## Conception d'une BDR

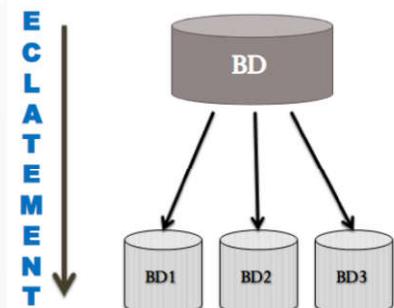
### ► Conception ascendante

- **(Bottom up design)** On utilise les différentes bases « locales » pour créer un schéma global.



### ► Conception descendante

- **(Top down design)** : On « éclate » le schéma global en **n bases locales**.



52

## Conception descendante

- ▶ La conception descendante est utilisée lors de la **construction d'une nouvelle base de données (Conception à partir du zéro)**.
- ▶ On créer d'abord **un schéma global**, les diverses entités sont alors **distribués sur les sites**.
- ▶ Les points suivants sont présents dans cette approche :
  - **Conception** du schéma conceptuel global
  - **Distribution** pour obtenir des schémas conceptuels locaux
  - Les tables du schéma global sont fragmentées (**processus de fragmentation**)
  - Les fragments sont donc placés sur des sites (**processus d'allocation**)

53

## Conception Ascendante

- ▶ La conception ascendante part de bases de données existantes.
- ▶ Obtention d'un schéma global à partir de schémas locaux.
- ▶ La **distribution est préexistante et doit être gérée**.
- ▶ Cette approche nécessite généralement une réconciliation sémantique (La réconciliation sémantique entre les contextes de deux SI permet d'importer les classes pertinentes et de les intégrer).
- ▶ Dans cette approche on part de l'existant (Distribution préexistante).
- ▶ L'objectif principal est d'intégrer les bases locales dans schéma global. Elle nécessite une consolidation, uniformisation c'est-à-dire :
  - **Réconciliation sémantique**
  - **Identifier les données semblables**
  - **Accorder leurs types, gérer leur cohérence...**
  - **Interfacer ou adapter les SGBD...**
- ▶ Exemple
  - Fusion de société

54

## Cas courant

- ▶ La mise en place de système distribué est généralement issu de conception ascendante
- ▶ C'est le cas le plus courant mais le plus compliqué, car il nécessite de gérer l'existant qui est généralement hétérogènes.
  - Machines (OS, Protocol,...)
  - SGBD
  - Modèles de données (réconciliation sémantique !!).