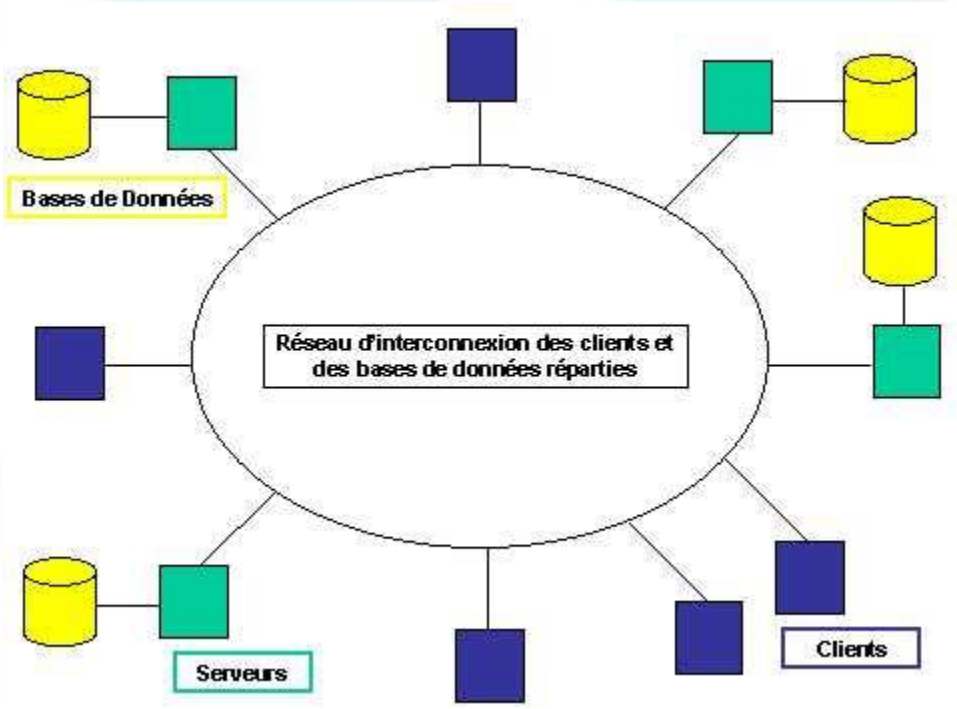
Bases de données réparties



EISTI

Quelques définitions

- DEFINITION : UNE BASE DE DONNEES REPARTIE EST UN ENSEMBLE DE DONNEES STOCKEES SUR PLUSIEURS SITES CONNECTES PAR UN RESEAU
- SITE : MACHINE AVEC UNE BDD LOCALE
- RESEAU : MOYEN DE COMMUNICATION ENTRE SITES
- CRITERES DE CHOIX : COUT INSTALLATION, COMMUNICATION, SURETE, DISPONIBILITE

- **Pour pouvoir gérer une base de données répartie, il faut pouvoir disposer d'un SGBD Réparti**

SGBD Réparti

- **Exécution des transactions**
  - **- *locales* : accès aux données sur site**
  - **- *globales* : accès sur plusieurs sites**
- **Définition : données locales/réparties**
- **Cohérence des données (semantic data controler)**
- **Contrôle de concurrence (transaction manager)**
- **Reprise après panne (recovery manager)**
- **Optimisation de requêtes (query processor)**
- **Indépendance des applications**
  - **- machine**
  - **- système d'exploitation**
  - **- protocole réseau**
  - **- système de gestion de bases de données**
  - **- localisation des données**

EISTI

Intérêts de la répartition des données

- Plus grande fiabilité (pannes des matériels)
- De meilleures performances
- Faciliter la croissance du système

EISTI

Inconvénients de la répartition des données

- Coûts (trafic réseau, hardware et software)
- Synchronisation pour accéder aux données
- Sécurité des données
- L’interblocage distribué

Architecture (I)

- Autonomie des sites
  - Intégration poussée
  - Semi-autonomie
  - Isolation totale
- Communication entre machines
  - Client-Serveur
    - Lourd : Le client se connecte à tous les serveurs de BDD
    - Léger : Le client ne se connecte qu'à un serveur (3 tiers)
  - Peer-to-Peer : toutes les machines ont le même rôle

Architecture (II)

- Conception de la base de données répartie
  - Méthode top-down design
    - Définition d'un schéma global : Global Conceptual Schema
    - Distribution vers des schémas locaux
    - Fragmentation et allocation des fragments
  - Méthode bottom-up design
    - Intégration de schémas locaux dans un schéma global

Les techniques de répartition

- La fragmentation
- La réplication

La Fragmentation

- **PRINCIPE**
  - **Répartir une relation (table) sur plusieurs sites. Sur chaque site est stockée une partie de la relation.**

- **TECHNIQUES**

Découpage Horizontal	⇔	**Sélection**
Découpage Vertical	⇔	**Projection**
Découpage Mixte	⇔	**Sélection et projection**

- **PROBLEME**
  - **Recomposer la relation initiale**

La Fragmentation Horizontale

- **On considère la relation R**
- **Att un attribut de la relation R**
- **$vAtt_1$, $vAtt_2$ …, $vAtt_n$, la liste exhaustive des valeurs de Att**
- **$\forall$ i $\in${1, .., n} on stocke sur le site si σ(R; Att = $vAtt_i$)**
- **$R = \cup_i \sigma(R; Att = vAtt_i)$**

La Fragmentation Verticale

- On considère la relation R
- $(Att_{1,1},\ldots,Att_{1,k1})$ … $(Att_{1,1},\ldots,Att_{1,kn})$ une liste d'ensembles d'attributs de R
- Chaque ensemble de la liste contient une clé de la relation R
- La réunion des ensembles recouvre tous les attributs de R
- Sur le site i, on stocke $\Pi(R; Att_{i,1},\ldots,Att_{i,ki})$

La Fragmentation Mixte

- 1) On considère une relation R
- 2) On la fragmente verticalement (projections)
- 3) Chaque fragment vertical peut être lui même décomposé en fragments horizontaux (sélections)

La réplication

- **PRINCIPE**
  - **Une relation stockée sur un site est recopiée sur un ou plusieurs sites**
- **AVANTAGES**
  - **Disponibilité des données**
  - **Augmentation du parallélisme en lecture**
  - **Diminution du coût imposé par les transmissions**
- **INCONVENIENTS**
  - **Cohérence des différentes copies**
  - **Propagation des mises a jour**

EISTI

La solution Oracle en informatique répartie

- **La couche Oracle Net**
  - Le fichier Listener.ora (fichier de configuration pour traiter les connexions distantes)
  - Le fichier tsname.ora (fichier contenant des alias de bases de données)
  - L'outil assistant Net Configuration pour mettre à jour les fichiers ci-dessus
  - Le programme listener pour détecter les requêtes de connexion
- **Le lien de base de données**
  - create [public] database link nomLien connect to user identified by passwd using 'nomService ' où nomService est un alias de bases de données défini sur le serveur de connexion

EISTI

La solution Oracle en informatique répartie

- **La réplication**
  - create snapshot image REFRESH FAST … AS SELECT …. FROM T@Lien;
- **La Défragmentation**
  - Création de vue avec une jointure sur plusieurs tables $T_1@L_1$, $T_n@L_n$

Validation d'une transaction

- Quoi
  - Une transaction T répartie $\Rightarrow$n transactions T_i locales
  - Il faut coordonner les sites S_i.
- Comment
  - Il faut un site coordinateur
  - On utilise la validation à deux phases

EISTI

Propriétés d'un protocole de validation

C'est un protocole de consensus sur l'exécution d'une transaction entre les participants qui émettent un vote

- Oui pour une validation (« commit »)
- Non pour un abandon (« rollback »)

1. La décision générale de validation est prise si tous les participants ont votés oui

2. Un participant ne peut plus revenir sur sa décision

- (voir http://cedric.cnam.fr/vertigo/Cours/Valeur-C/Exposes/concurrence.ppt)

EISTI

Le protocole de validation à deux phases

- Phase 1:
  - Le coordonateur ***C*** écrit <T préparée> dans son journal
  - ***C*** envoie « préparation » à tous les sites S_i
  - Chaque site $\mathbf{S_i}$ prend une décision pour T_i :
    - s'il valide T_i , alors :
      - $\mathbf{S_i}$ écrit <T prête > dans son journal
      - $\mathbf{S_i}$ envoie « T prête » au coordonateur ***C***
    - s'il ne valide pas T_i , alors :
      - $\mathbf{S_i}$ écrit <non T> dans son journal
      - $\mathbf{S_i}$ envoie « abandon T » au coordonnateur ***C***

- (voir http://cedric.cnam.fr/vertigo/Cours/Valeur-C/Exposes/concurrence.ppt)

Le protocole de validation à deux phases

- PHASE 2
  - A partir des messages reçus des sites S_i, ***C*** prend une décision :
    - S'il a reçu « T prête » de tous les sites S_i :
      - il ajoute <T validée> à son journal
      - il envoie « valider T » à tous les sites S_i
    - S'il a reçu au moins un message « abandon T » d'un site S_i :
      - il ajoute <T abandonnée> à son journal
      - il envoie <abandon T> à tous les sites S_i
  - Chaque site **Si** :
    - inscrit dans son journal le message envoyé par **C** (<T validée> ou <T abandonnée>)
    - envoie un accusé de réception au coordinateur **C**
  - Quand le coordinateur ***C*** a reçu tous les accusés des sites $\mathbf{S_i}$, il ajoute : <T finie> à son journal
- (voir http://cedric.cnam.fr/vertigo/Cours/Valeur-C/Exposes/concurrence.ppt)

Traitement de reprise après une panne

- Après panne du coordinateur :
  - Si reprise dans la phase préparatoire avant diffusion du statut de la transaction => abandonner la transaction.
  - Si reprise après décision prise => ne rien faire.

- Après panne chez un exécutant :
  - Si reprise dans la phase préparatoire => abandonner la transaction.
  - Si reprise en état de décision connue => ne rien faire.

- (voir http://cedric.cnam.fr/vertigo/Cours/Valeur-C/Exposes/concurrence.ppt)