Sécurité des systèmes d'information

Mehdi Haddad
mehdi.haddad@u-pec.fr

2017 - 2018

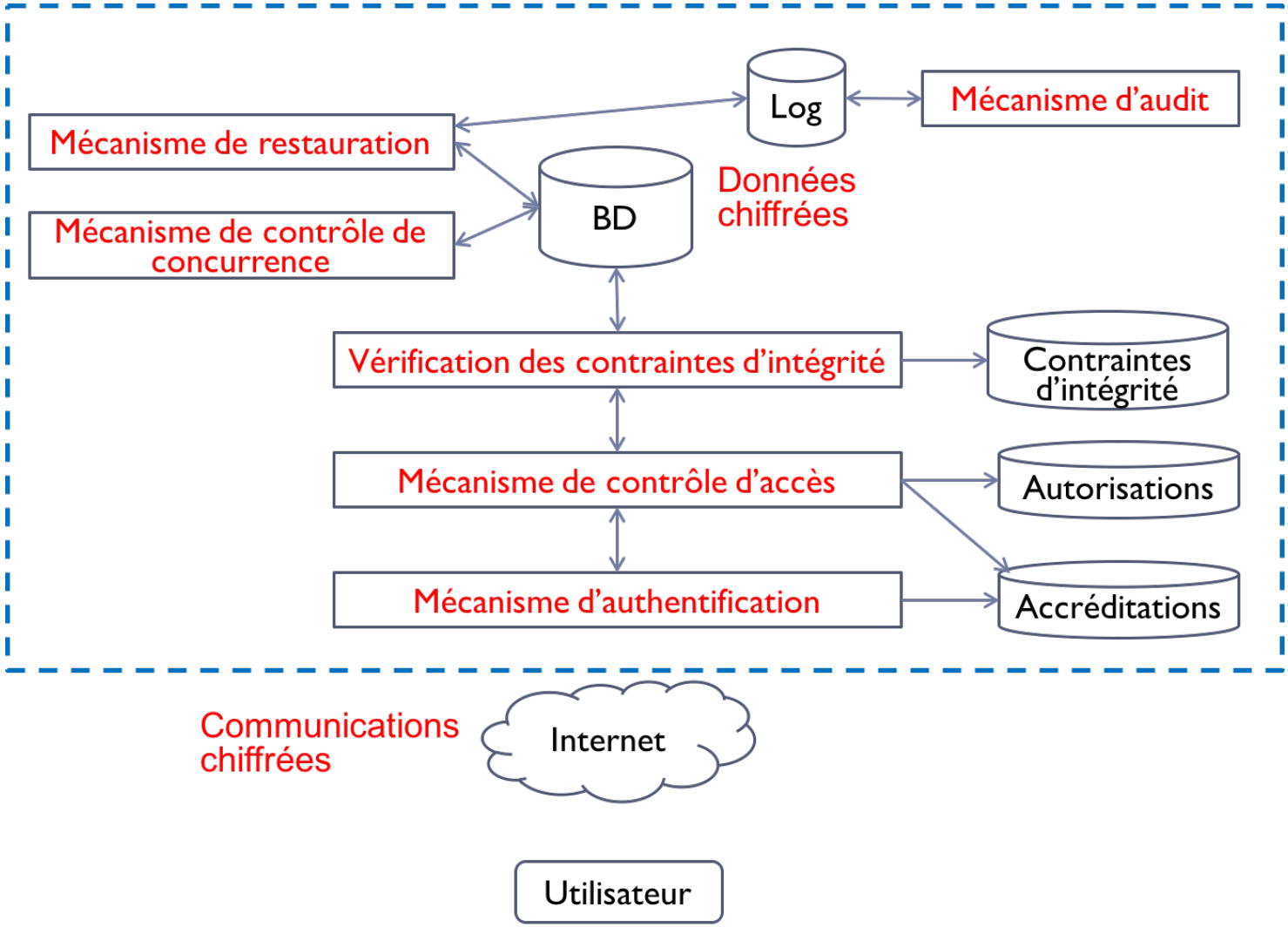
Sécurité des bases de données

- Ensemble des mécanismes qui protège la base de données contre des menaces intentionnelles ou accidentelles
- Remarque : il est nécessaire de prendre en compte l'environnement de la BD, c-à-d le réseau, le matériel, le logiciel, ...

Propriétés désirées

- Confidentialité
  - Protection de l'information contre la divulgation non autorisée
  - L'information n'est accessible que par les personnes autorisées
- Intégrité
  - Protection de l'information contre des modifications
  - authenticité : assurer l'intégrité de l'information par rapport à l'information originale
  - protéger l'information contre des modifications non autorisées
  - intégrité sémantique : protéger l'information contre des modifications incorrectes
- Disponibilité
  - Prévention de refus d'accès autorisé à des informations ou des ressources
  - Les utilisateurs autorisés doivent pouvoir accéder à l'information lorsqu'ils en ont besoin

Sécurité dans les base de données



Liens entre les mécanismes et propriétés désirées

Authentification

Contrôle d'accès

Audit

Cryptographie

Restauration

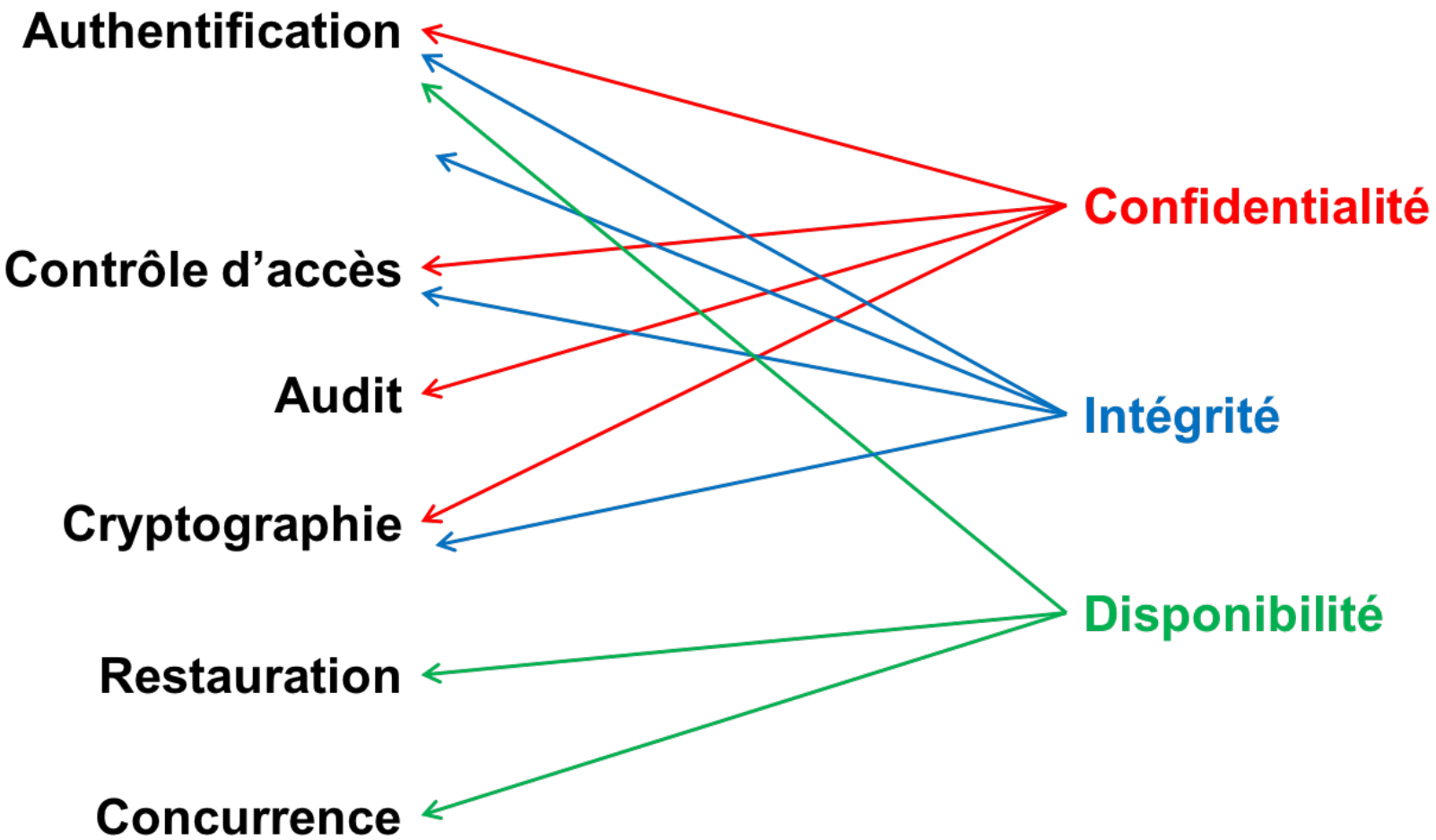
Concurrence

Confidentialité

Intégrité

Disponibilité

Liens entre les mécanismes et propriétés désirées



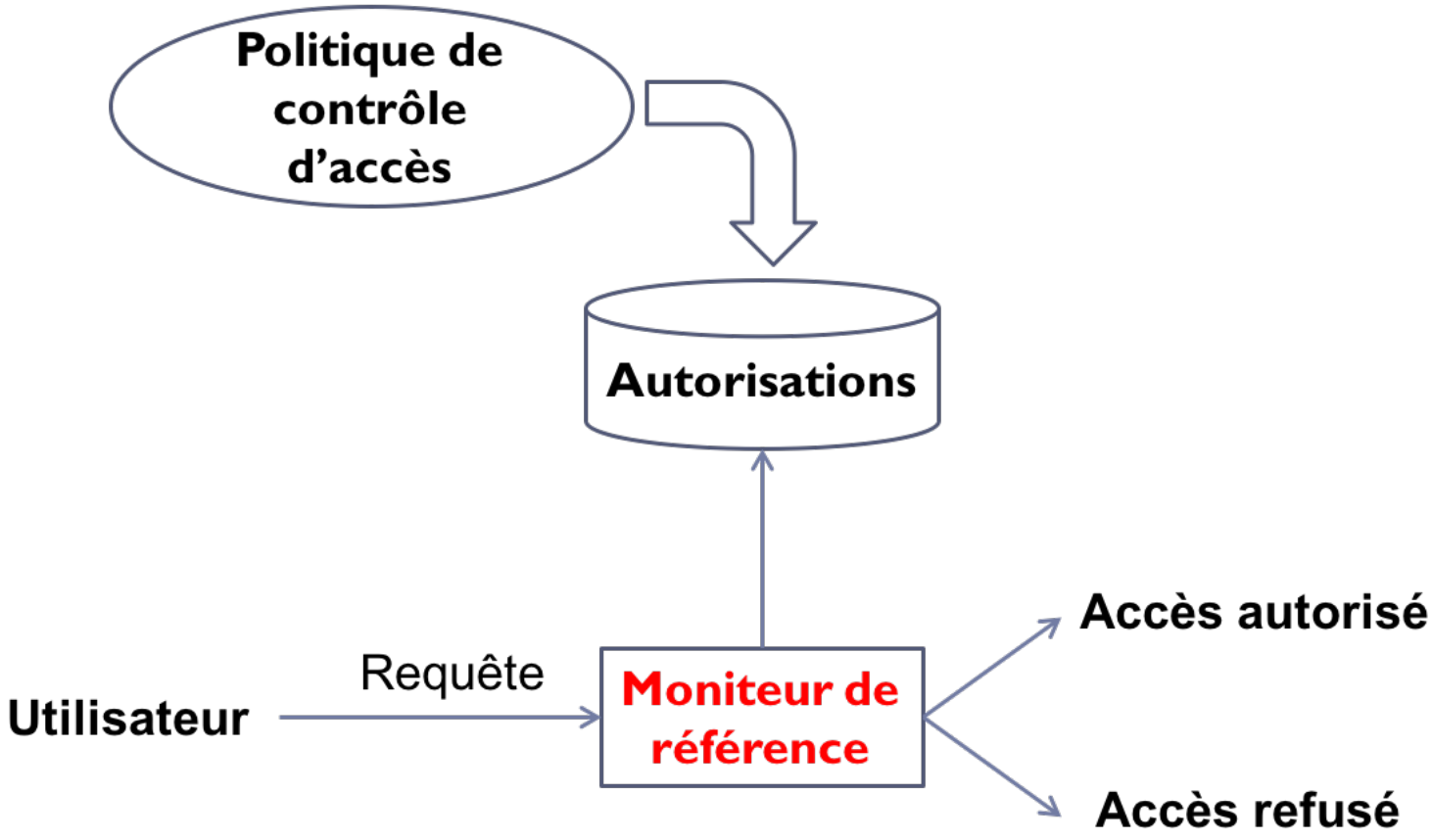
Contrôle d'accès

- Composant majeur de la sécurité des bases de données
- Différents niveaux d'abstraction pour le développement d'un système de contrôle d'accès
  - Politique de sécurité
  - Modèle de contrôle d'accès
  - Mécanisme de contrôle d'accès

Objectifs du contrôle d'accès

- Le contrôle d'accès a pour objectif d'empêcher que des opérations non autorisées puissent être effectuées sur les objets
- Le contrôle d'accès est souvent régi par un ensemble d'autorisations définies par le politique de sécurité
- Les autorisations sont mises en œuvre par un moniteur de référence

Processus général du contrôle d'accès

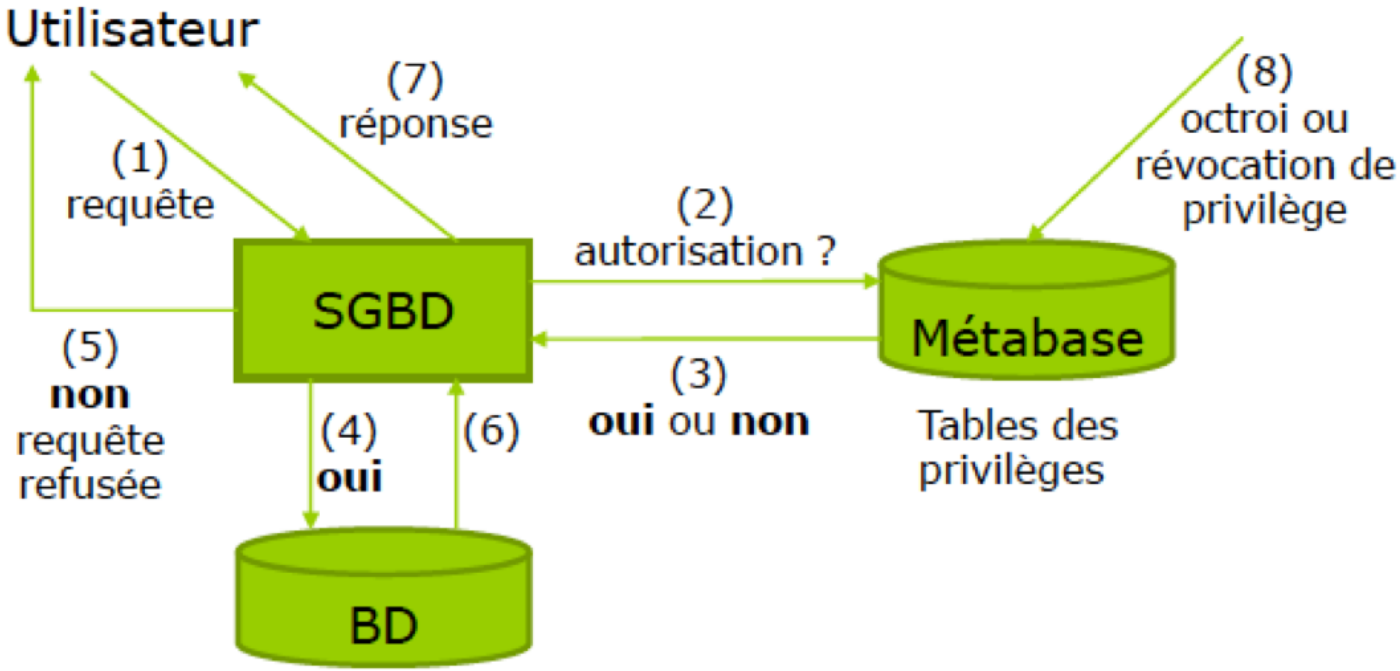


Gestion des autorisations en SQL

Vocabulaire du contrôle d'accès dans un contexte BD

- Sujets : Utilisateur ou Rôle
- Objets : BD, table, vue, etc
- Privilèges : Select, Update, Insert, Delete, etc

Contrôle d'accès dans un SGBD



Gestion des utilisateurs

- L'administrateur de base de données peut créer des utilisateurs
- Les utilisateurs peuvent se connecter à la base de données et acquérir des privilèges sur les objets
- Pas de syntaxe commune à tous les SGBD commerciaux

Privilèges

On peut distinguer deux types de privilèges :

- Les privilèges sur les objets
  - Opérations de manipulation sur les tables, les vues ou les procédures
  - Select, update, insert, delete
- Les privilèges d'administration
  - Création d'une base, d'une table, d'utilisateurs, etc

  Syntaxe spécifique à chaque SGBD

Privilèges objet

- SELECT ou SELECT(c1, . . . , cn) : pour pouvoir lire le contenu de toutes ou de certaines colonnes d'une table
- INSERT ou INSERT(c1, . . . , cn) : pour pouvoir insérer une valeur dans toutes ou certaines colonnes d'une table
- UPDATE ou UPDATE(c1, . . . , cn) : pour pouvoir modifier le contenu de toutes ou de certaines colonnes d'une table
- DELETE : pour pouvoir supprimer des lignes d'une table

Privilèges objet

- REFERENCES REFERENCES(c1, . . . , cn) : pour pouvoir faire référence à une table ou à certaines colonnes d'une table dans une contrainte d'intégrité
- TRIGGER : pour pouvoir placer un trigger sur une table
- EXECUTE : pour pouvoir exécuter une procédure stockée.

Gestion des privilèges

SQL définit deux commandes pour octroyer ou révoquer des privilèges

- GRANT
- REVOKE

Commande GRANT

- Syntaxe :

```
GRANT (liste de privilèges | ALL)
ON liste d'objets
TO liste d'utilisateurs | PUBLIC
[WITH GRANT OPTION]
```

- ALL : tous les privilèges que le donneur peut accorder
- PUBLIC : tous les utilisateurs connus du système
- WITH GRANT OPTION : indique que le receveur pourra transmettre les privilèges qui lui sont octroyés

Commande REVOKE

- Syntaxe :

```
REVOKE [GRANT OPTION FOR] (privilèges | ALL)
ON liste d'objets
FROM liste d'utilisateurs
[{RESTRICT | CASCADE}]
```

- CASCADE : la révocation concerne les utilisateurs cités dans la clause FROM ainsi que ceux à qui ces privilèges ont été récursivement transmis
- RESTRICT : la révocation ne concerne que les utilisateurs cités dans la clause FROM
- GRANT OPTION FOR : ce n'est pas les privilèges qui sont révoqués, mais le droit de le transmettre

Règles d'octroi des privilèges

- Le créateur d'un objet possède tous les privilèges sur cet objet
- Un utilisateur ne peut transmettre que les privilèges qu'il possède
- Si l'option CASCADE est spécifiée, la révocation d'un privilège est récursive.
- Si l'option RESTRICT est spécifiée, la révocation d'un privilège à un utilisateur n'est possible que si celui-ci n'a pas transmis ce privilège à un autre utilisateur.
- Si un utilisateur a reçu un privilège de plusieurs utilisateurs, il ne perd ce privilège que si tous ces utilisateurs le lui retirent.

Exécution d'un Grant

- Le SGBD garde, pour chaque utilisateur, les privilèges qu'il possède et ceux qu'il peut transmettre.
- A chaque fois qu'un utilisateur exécute une commande Grant, le système fait l'intersection entre les privilèges qu'il peut accorder et l'ensemble des privilèges spécifiés dans la commande. Si l'intersection est vide, la commande n'est pas exécutée.

Exécution d'un Grant : exemple

- Bob : GRANT select, insert ON Employee TO Jim WITH GRANT OPTION ;
- Bob : GRANT select ON Employee TO Ann WITH GRANT OPTION ;
- Bob : GRANT insert ON Employee TO Ann ;
- Jim : GRANT update ON Employee TO Tim WITH GRANT OPTION ;
- Ann : GRANT select, insert ON Employee TO Tim ;

Exécution d'un Grant : exemple

- Les 3 premiers GRANT sont exécutés (Bob est le propriétaire de la relation)
- Le 4ème GRANT n'est pas exécuté car Jim n'a pas de privilège update sur la relation.
- Le 5ème GRANT est partiellement exécuté : Ann a les privilèges select et insert mais pas le grant option pour le insert.
- Tim reçoit seulement le privilège select

Commande Revoke : exemple

- Bob : GRANT select ON Employee TO Jim WITH GRANT OPTION ;
- Bob : GRANT select ON Employee TO Ann WITH GRANT OPTION ;
- Jim : GRANT select ON Employee TO Tim ;
- Ann : GRANT select ON Employee TO Tim ;
- Jim : REVOKE select ON Employee FROM Tim ;

Tim continue de posséder le privilège select sur Employee après la commande car il l'a reçu aussi de Ann indépendamment.

Vues et autorisations

- Les vues sont utilisées pour gérer le contrôle d'accès basé sur le contenu :
- Définir une vue V contenant les tuples qui peuvent être lus par un sujet S
- Accorder à S le privilège "select" sur la vue V, et pas sur les relations de base.

Vues et autorisations : exemple

- Supposons qu'on veuille autoriser l'utilisateur Ann à accéder seulement aux employés dont le salaire est inférieur à 20000 euros :
- ```
  CREATE VIEW Vemp AS
  SELECT * FROM Employee
  WHERE Salary < 20000;
  ```
- ```
  GRANT Select ON Vemp TO Ann;
  ```

Exécution de requêtes sur les vues

- Les requêtes sur les vues sont transformées par le mécanisme de composition de vues en requêtes sur les relations de la BD
- Le mécanisme de composition de vues combine par un AND les prédicats qui définissent la vue et les prédicats présents dans la requête.

Exécution de requêtes sur les vues

- ```
  CREATE VIEW Vemp AS
  SELECT * FROM Employee
  WHERE Salary < 20000;
  ```
- ```
  SELECT * FROM Vemp
  WHERE Job = 'Programmer';
  ```
- Requête après composition :

  ```
  SELECT * FROM Employee
  WHERE Salary < 20000
  AND  Job = 'Programmer';
  ```

Exécution de requêtes sur les vues

- Attribution du privilège Select pour des statistiques :
- `Employee (Num, Name, Address, NumDepart, Salary)`
  `Department (NumDepart, Head, Location)`

- ```
  CREATE VIEW AvgSal (NumDepart, AvgSalary) AS
   SELECT NumDepart, Avg(all Salary)
       FROM Department natural join Employee
          GROUP BY NumDepart;
  ```

- `GRANT Select ON AvgSal TO Ann;`
  ```

Modification de vues

- Modifications de vues Insert, Update, Delete
- Les modifications apportées à V sont réécrites pour modifier les tables d'origines

Modification de vues

- Deux manières d'effectuer ces modifications
  - Réécriture spécifiée par le créateur de la vue en utilisant des déclancheurs (TRIGGER)
    - avantage : peut gérer toutes les modifications
    - inconvénient : pas de de garantit concernant l'intégrité des donnée
  - Réécriture automatique par le SGBD : implique de restreindre
    - avantage : pas d'intervension manuelle de l'utilisateur
    - inconvénient : les restrictions sont assez importantes
- Modification de vue impossible dans certains cas. Exemple :

```
GRANT update  ON AvgSal TO Ann;
```

Modification de vues

- Restriction dans le strandard SQL "updatable view"
- Une vue est modifiable si elle respecte les 4 restrictions suivantes :
  1) utilise un SELECT (sans clause DISTINCT) sur une seule table T
  2) Les attributs qui n'appartiennent pas à la vue peuvent être NULL ou ont une valeur par défaut
  3) Les sous requêtes ne peuvent pas utiliser T
  4) Pas de Group By ou d'agrégation
- Les SQBD commerciaux peuvent être plus flexible de que le standard Exemple : Oracle permet, sous certaines conditions, la modification d'une vue définie par une jointure

Autorisation sur une vue

- Une autorisation de sélection (grant select) peut s'appliquer à n'importe quelle vue sans restriction concernant de sa définition
- Une autorisation portant sur une modification (grant insert, grant update ou grant delete) ne peut s'appliquer que sur une vue modifiable