C018SA-W4-S3

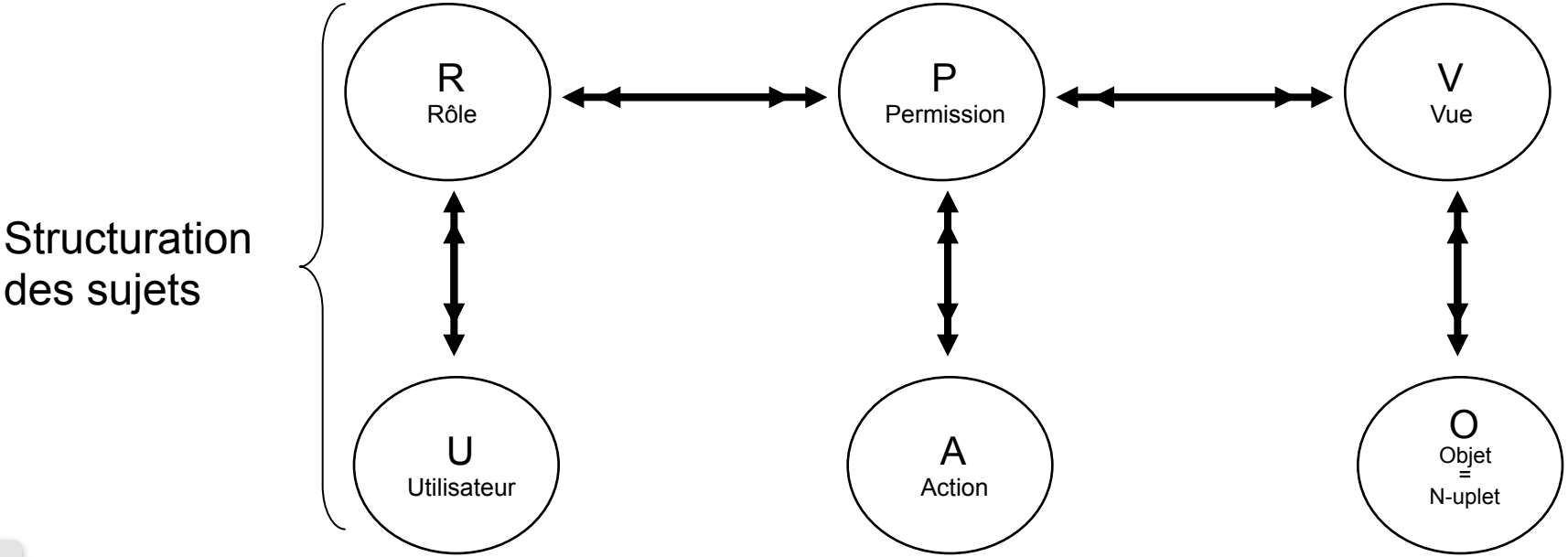
informatiques mathématiques

SEMAINE 4 : Contrôle d'Accès

1. Introduction
2. Modèle de contrôle d'accès discrétionnaire (DAC)
3. **Modèle de contrôle d'accès basé sur les rôles (RBAC)**
4. Modèle de contrôle d'accès obligatoire (MAC)

Le modèle RBAC en SQL (99)

- **Rôle =** ensemble de privilèges
- Les accès des utilisateurs sont gérés en fonction de leur rôle organisationnel
- **Objectif =** faciliter l'administration des droits



Opérations de gestion des rôles

```
CREATE ROLE <nom_role> ;
```

- Création d'un nouveau rôle nom_role

```
DROP ROLE <nom_role> ;
```

- Suppression du rôle nom_role

```
SET ROLE <liste_roles> ;
```

- Permet à un utilisateur d'activer un ensemble de rôles pendant la durée d'une session SQL

Evolution de l'instruction GRANT

- Affectation des privilèges aux rôles

```
GRANT <liste privileges>
ON <table ou vue>
TO <liste roles>
[ WITH GRANT OPTION ] ;
```

- Affectation des rôles aux utilisateurs

```
GRANT <liste roles>
TO <liste utilisateurs> ;
```

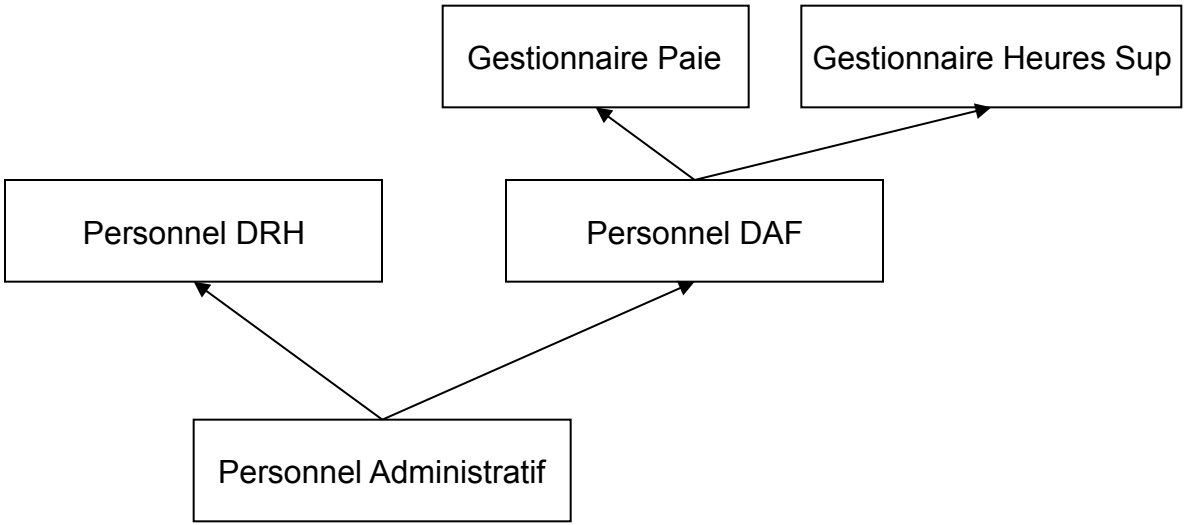
- Rôle junior et rôle senior

```
GRANT <r1> TO <r2> ;
```

Le rôle *r2* reçoit tous les privilèges du rôle *r1*

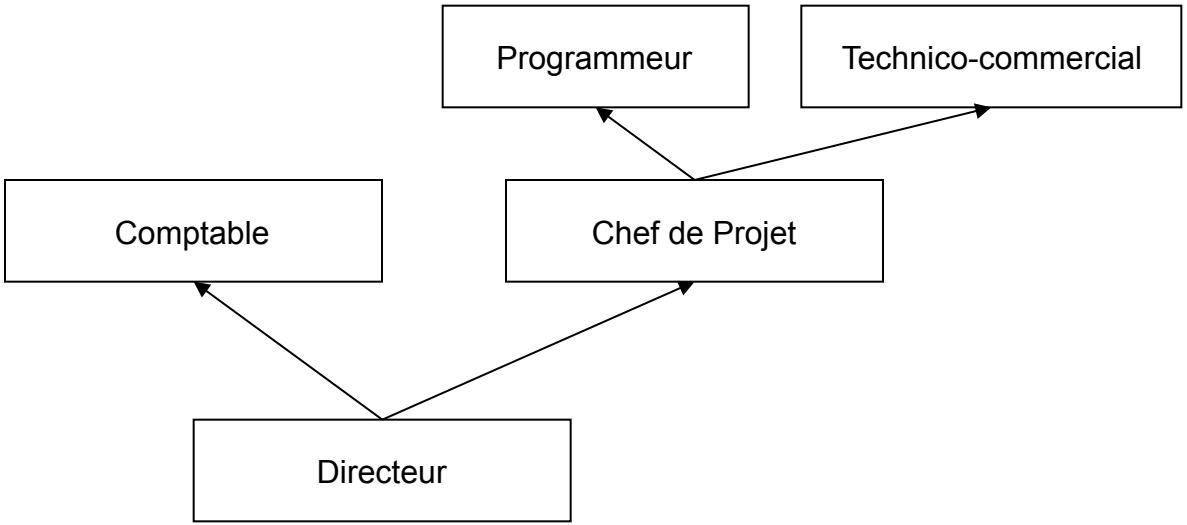
Hiérachie des rôles

- **Spécialisation/Généralisation**
  - *r1* est un rôle senior de *r2* si chaque fois qu'un utilisateur joue le rôle *r1*, cet utilisateur joue aussi le rôle *r2*
  - Les feuilles de la hiérarchie ont plus de privilèges que la racine



Hiérachie des rôles

- **Hiérarchie organisationnelle**
  - *r1* est un rôle senior de *r2* si un utilisateur jouant le rôle *r1* est un supérieur hiérarchique d'un utilisateur jouant le rôle *r2*
  - Les feuilles de la hiérarchie ont moins de privilèges que la racine



Les contraintes

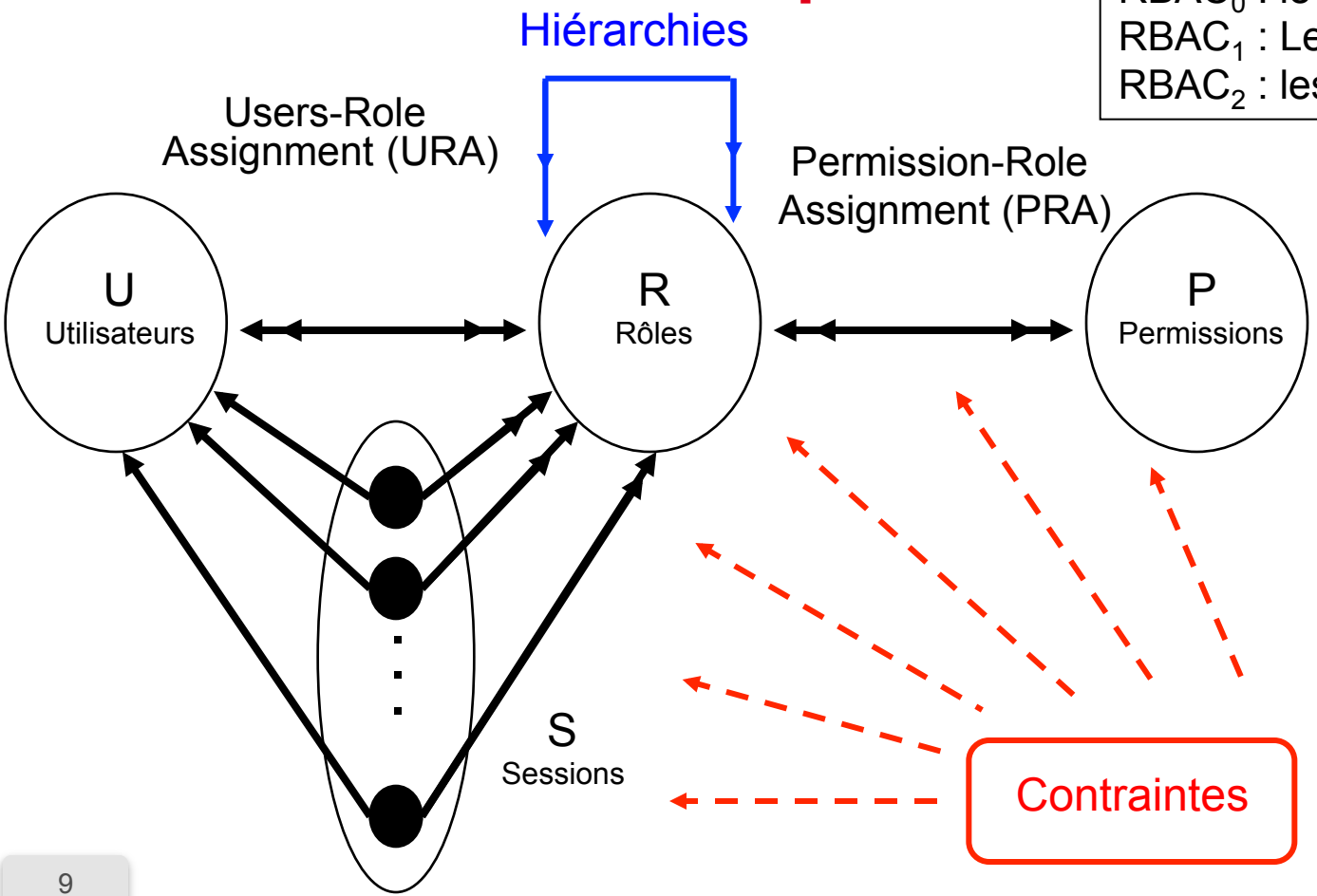
- **Contrainte sur Utilisateur / Rôle**
  - Contrainte de type « Séparation des pouvoirs »
  - *Exemple :* rôles comptable et chef de projet sont exclusifs

- **Contrainte sur Session / Rôle**
  - Un utilisateur peut cumuler plusieurs rôles mais pas les activer dans une même session
  - Contrainte de type « Séparation des tâches »

- **Contrainte sur Rôle / Permission**
  - Un rôle ne doit pas pouvoir cumuler certaines Permissions
  - *Exemple :* une action + l'audit ou la validation de cette action
  - Autre contrainte de type « Séparation des tâches »

Le modèle RBAC complet

$RBAC_0$: le noyau (URA + PRA)
$RBAC_1$: Les hiérarchies
$RBAC_2$: les contraintes



Vers ABAC (Attribute Based Access Control)

- Utilisation des valeurs d'un attribut pour définir les droits
  e.g. attribut d'un utilisateur : une vidéo n'est visionnable que si l'utilisateur a plus de 18 ans

- On peut utiliser les attributs des utilisateurs, des ressources, des actions et des contextes

- Intégré depuis 2010 au modèle standard RBAC du NIST