



M2106

Programmation et administration
des bases de données

COURS 3 : Administration de bases de données

Avril, 2017

BELFODIL Adnene – BENDIMERAD Anes



- Création et suppression d'utilisateurs
- Octroi et révocation de privilèges
- Regroupement de privilèges : les rôles
- Méta-schéma

- Création et suppression d'utilisateurs
- Octroi et révocation de privilèges
- Regroupement de privilèges : les rôles
- Méta-schéma

Syntaxe de création d'un utilisateur

```
CREATE USER nom_utilisateur  
DEFAULT TABLESPACE nom_table_space  
IDENTIFIED BY mot_de_passe  
QUOTA {UNLIMITED | tailleEnOctet} ON nom_table_space;
```

Syntaxe de suppression d'un utilisateur

```
DROP USER user_name [ CASCADE ];
```

Exemple

```
-- création :  
CREATE USER bernard  
DEFAULT TABLESPACE users  
IDENTIFIED BY mdpBernard  
QUOTA 100 M ON users;  
-- vérifier que le compte est créé :  
SELECT * FROM all_users WHERE USERNAME='bernard';  
-- suppression :  
DROP USER bernard
```

- Création et suppression d'utilisateurs
- **Octroi et révocation de privilèges**
- Regroupement de privilèges : les rôles
- Méta-schéma

Définition

Les privilèges consistent en l'ensemble des opérations autorisées pour chaque utilisateur. Ils permettent de protéger les données en accordant (ou retirant) des droits différents à chaque utilisateur ou groupe d'utilisateurs.

Chaque utilisateur dispose des privilèges qui lui sont octroyés avec la commande **GRANT**. Par ailleurs, un utilisateur peut se voir révoquer des privilèges en utilisant la commande **REVOKE**.

GRANT : Octroi de privilèges

```
-- octroie de privilèges à l'utilisateur 'nom_utilisateur'  
GRANT {ALL PRIVILEGES | priv1,..., privN } TO nom_utilisateur;
```

REVOKE : Révocation de privilèges

```
-- révoque des privilèges à l'utilisateur 'nom_utilisateur'  
REVOKE {ALL PRIVILEGES | priv1,..., privN } FROM nom_utilisateur;
```


Exemple d'octroi de privilèges

```
-- Exécuté en étant connecté en tant que système
-- octroie les droits de connexion et de création de tables et de vues
GRANT CREATE SESSION, CREATE TABLE, CREATE VIEW TO bertrand;

-- Exécuté en étant connecté à martin (propriétaire) ou à système
-- autorise bertrand à interroger la table martin.movie
GRANT SELECT ON martin.movie TO bertrand;

-- autorise bertrand à ajouter des lignes dans martin.categorie
GRANT INSERT ON martin.categorie TO bertrand;
```

Exemple de révocation de privilèges

```
-- Exécuté en étant connecté en tant que système
-- L'utilisateur bertrand ne pourra plus créer des vues
REVOKE CREATE VIEW FROM bertrand;

-- Exécuté en étant connecté à martin (propriétaire) ou à système
-- Annuler l'autorisation d'interroger la table martin.movie
REVOKE SELECT ON martin.movie FROM bertrand;

-- Annuler l'autorisation d'ajouter des lignes dans martin.categorie
REVOKE INSERT ON martin.categorie FROM bertrand;
```

- Création et suppression d'utilisateurs
- Octroi et révocation de privilèges
- Regroupement de privilèges : les rôles
- Méta-schéma

Définition

Un rôle est un regroupement de privilèges. Une fois créé, il peut être assigné à un utilisateur.

Syntaxe

```
-- définition d'un rôle
CREATE ROLE nom_role;
-- octroie des privilèges au role 'nom_role'
GRANT {ALL PRIVILEGES | priv1,...,privN } TO nom_role;
-- octroie du rôle 'nom_role' à l'utilisateur 'nom_utilisateur'
GRANT nom_role TO nom_utilisateur
```

Exemple 1 : création et attribution d'un rôle

```
-- Création d'un rôle et octroi de privilèges à ce rôle
CREATE ROLE r_etudiant;
GRANT CREATE SESSION, CREATE TABLE, CREATE VIEW TO r_etudiant;
-- Création d'un utilisateur et octroi du rôle
CREATE USER bertrand ...;
GRANT r_etudiant TO bertrand;
-- Avantage des rôles : modification répercutée sur tous les utilisateurs
GRANT CREATE SEQUENCE TO r_etudiant;
REVOKE CREATE VIEW FROM r_etudiant;
```

Exemple 2 : création d'un rôle par extension d'un rôle existant

```
-- Un autre avantage des rôles : héritage des privilèges par création
-- hiérarchique des rôles
CREATE ROLE r_enseignant;
GRANT r_etudiant, CREATE PROCEDURE TO r_enseignant;

CREATE USER martin ...;
GRANT r_enseignant TO martin;
```

Consultation des rôles et privilèges d'un utilisateur

```
-- Consultation des rôles octroyés au compte courant  
SELECT * FROM user_role_privs;  
-- Consultation des privilèges octroyés au compte courant  
SELECT * FROM user_tab_privs_recd;  
-- Consultation des privilèges octroyés par compte courant  
SELECT * FROM user_tab_privs_made;
```

Remarque

Le rôle **dba** (database administrator) rassemble tous les privilèges

- Création et suppression d'utilisateurs
- Octroi et révocation de privilèges
- Regroupement de privilèges : les rôles
- Méta-schéma

Définition

Le méta-schéma du SGBD mémorise des informations sur les objets qu'il gère : tablespaces, utilisateurs, tables, vues, procédures, etc. Il comprend des vues qui sont mises à jour automatiquement (exemple : suite à une création de table). Ces vues préfixées « all_ », « dba_ » et « user_ » sont interrogeables par les utilisateurs.

Exemple : occupation disque des objets du schéma courant ?

```
-- Interrogation de la vue méta-schéma 'user_segments' depuis le compte  
-- 'martin'
```

```
SELECT segment_name, tablespace_name, bytes/1024/1024 Mo  
FROM user_segments ORDER BY Mo DESC, segment_name;
```

-- SEGMENT_NAME	TABLESPACE_NAME	Mo

-- NDX_INDEXTITLE_IDTERM	martin	240
-- PK_DOC	martin	150
-- ...		

Types de vues du méta-schéma (3 types)

1. Vues prefixées par « `dba_` » : Elles listent les informations sur tous les objets de la BD. Elles sont interrogeables seulement par les administrateurs. Exemple : `dba_tables` liste toutes les table de la BD.
2. Vues préfixées par « `all_` » : Elles listent les informations sur les objets accessibles par l'utilisateur courant. Exemple : `all_tables` liste les tables que l'utilisateur peut manipuler.
3. Vues préfixées par « `user_` » : Elles listent les informations sur les objets possédés par l'utilisateur courant. Exemple : `user_tables` liste les tables possédées par l'utilisateur courant.

Table ou vue	Description
<code>all_catalog</code>	Liste des objets accessibles par l'utilisateur courant
<code>all_users</code>	Liste des utilisateurs créés sur l'instance courante
<code>user_segments</code>	Informations sur l'espace disque occupé par les objets
<code>user_ts_quotas</code>	Quotas fixés sur les tablespaces de l'utilisateur courant
<code>user_objects</code>	Objets créés par l'utilisateur courant
<code>user_tables</code>	Tables créées par l'utilisateur courant
<code>user_tab_columns</code>	Colonnes des tables créées par l'utilisateur courant
<code>User_constraints</code>	Contraintes créées sur des tables de l'utilisateur courant
<code>User_cons_columns</code>	Colonnes ciblées par les contraintes créées sur des tables

Table ou vue	Description
user_procedures	Procédures et fonctions créées par l'utilisateur courant
user_triggers	Déclencheurs créés par l'utilisateur courant
user_views	Vues créées par l'utilisateur courant
user_sys_privs	Privilèges « système » octroyés à l'utilisateur courant
user_tab_privs	Privilèges « objet » octroyés à l'utilisateur courant
user_tab_privs_recd	Privilèges octroyés au compte courant par d'autres
user_tab_privs_made	Privilèges octroyés par le compte courant à d'autres comptes
user_role_privs	Rôles octroyés à l'utilisateur courant
role_sys_privs	Privilèges octroyés au rôles