COURS ADMINISTRATION DE BASES DE DONNÉES PARTIE 4

Karim LABIDI
ISET Ch
2015-2016

Contenu

- La gestion des utilisateurs
- La gestion des profils
- La gestion des roles

Gestion des utilisateurs

- Un compte utilisateur = Un schéma de BD
  - C'est un ensemble d'objets : tables, vues, index,...
  - L'utilisateur créer modifie, ou supprime ses objets
- On peut créer des utilisateurs "simples"
  - Droits de requêtes sur un schéma précis
  - Pas de droits de créations

Gestion des utilisateurs

- Les profils:

  chaque utilisateur peut voir sa consommation de ressources système contrôlée par un profil spécifique

- Les privilèges d'accès:
  - privilèges système
  - privilèges objets
- Les rôles:

  chaque utilisateur peut voir ses privilèges définis par un rôle particulier.

Création d'un utilisateur de la base

- A chaque création d'un **utilisateur** correspond la création d'un **schéma** de même nom
- Un **mot de passe** est donné à chaque utilisateur
- CREATE USER *utilisateur*

  IDENTIFIED BY *mot_de_passe*/EXTERNALLY

  [DEFAULT TABLESPACE *tablespace*]

  [TEMPORARY TABLESPACE *tablespace*]

  [PROFILE *profil*]

  {QUOTA *entier*/UNLIMITED ON *tablespace*}

Création d'un utilisateur de la base

- Exemple:

```
CREATE USER iset
    IDENTIFIED BY isetch2015
    DEFAULT TABLESPACE users
    TEMPORARY TABLESPACE temp
    QUOTA 22M ON users
    /*pas besoin de quotas sur temp*/
```

Les profils

- Ensembles nommés de limites ressources
- Affectés à des utilisateurs

  (il ne peut être attribué à chaque utilisateur qu'un seul profil)
- Activés ou désactivés:
  - paramêtre RESOURCE_LIMIT dans init.ora (TRUE ou FALSE)
  - ALTER SYSTEM

    SET RESOURCE_LIMIT = TRUE / FALSE

Les profils

- Avantages:
  - permettent de contrôler les ressources systèmes de chaque utilisateur:
    - interdire des opérations coûteuses
    - s'assurer que les utilisateurs se déconnectent
    - grouper des limites pour des groupes d'utilisateurs
  - sont utiles sur de gros systèmes pour contôler l'utilisation des ressources d'un groupe d'utilisateurs par rapport à un autre
- Inconvénients:
  - charges supplémentaires pour le contrôle de la limite autorisée de chaque utilisateur

Profils et Ressources

Ressource	Description
SESSIONS_PER_USER	nombre maximal de sessions concurrentes autorisées
CPU_PER_SESSION	temps CPU maximal par session en centième de sec.
CPU_PER_CALL	temps CPU maximal pour un appel noyau
CONNECT_TIME	temps de connexion écoulé exprimé en minutes
IDLE_TIME	temps d'inactivité continue exprimé en minutes
LOGICAL_READ_PER_ SESSION	nombre maximal de blocs de données lus par session
LOGICAL_READ_PER_CALL	nombre maximal de blocs de données lus par appel
COMPOSITE_LIMIT	coût total des ressources pour une session
PRIVATE_SGA	taille maximale (en K ou M) allouée à la SGA (MTS)

Création, modification et suppression d'un profil

- Création et modification d'un profil:

CREATE / ALTER PROFILE *nom_du_profil*

LIMIT {*ressource entier* / UNLIMITED / DEFAULT}

- Il existe un profil DEFAULT créé à la création de la base.

- Suppression d'un profil:

DROP PROFILE *nom_du_profil* [CASCADE]

- CASCADE permet de supprimer le profil pour tous les utilisateurs concernés et de leur affecter le profil DEFAULT.

Création d'un profil

- Exemple:

```
CREATE PROFILE developpeur_forms LIMIT
    SESSIONS_PER_USER              7
    CPU_PER_SESSION                UNLIMITED
    IDLE_TIME                      30
    COMPOSITE_LIMIT                1500000;
```

Quand une limite de ressource est atteinte pour un utilisateur, le SGBD arrête l'exécution de l'opération en cours et annule la transaction.

COMPOSITE_LIMIT RESOURCE COST

- COMPOSITE_LIMIT exprime, pour une session, le coût total des ressources c_*limit*:
  - CPU_PER_SESSION
  - CONNECT_TIME
  - LOGICAL_READ_PER_ SESSION
  - PRIVATE_SGA

- Ce coût est positionné par la commande:

  ALTER RESOURCE COST {c_*limit entier*}

COMPOSITE_LIMIT RESOURCE COST

- Exemple:

ALTER RESOURCE COST

CPU_PER_SESSION 150

CONNECT_TIME 2;

La formule de calcul de ce coût est:

T=150*CPU+2*CON

Le poids par défaut pour une ressource c_*limit* est 0.

Profil par défaut

- Les utilisateurs qui n'ont pas explicitement un profil ont le profil DEFAULT
- Toutes les limites non spécifiées d'un profil ont les valeurs du profil DEFAULT
- Initialement toutes les valeurs par défaut sont illimitées
- Le profil DEFAULT peut être modifié

Vues du dictionnaire et profils

- DBA_USERS: permet d'obtenir des informations sur tous les utilisateurs et leur profil associé
- USER_RESOURCE_LIMITS: décrit les ressources limites de l'utilisateur courant
- DBA_PROFILES:décrit les ressources limites de chaque profil
- RESOURCE_COST: décrit le poids de chaque ressource de COMPOSITE_LIMIT

Les privilèges système

- Un privilège système permet d'exécuter certaines actions touchant la structure de la base:
  - créer une session (une connexion)
  - créer une table ou une séquence etc...:
    - sur son propre schéma
    - sur tout autre schéma,
  - créer un profil, un utilisateur etc...

Attribution de privilèges système

- GRANT *priv._syst. / rôle {, priv._syst. / rôle}* TO *user / rôle {,user/ rôle}* / PUBLIC
  [WITH ADMIN OPTION]
  - WITH ADMIN OPTION permet aux destinataires de transmettre les privilèges ou les rôles à d'autres utilisateurs ou à d'autres rôles.
  - Retirer les privilèges système à un utilisateur ne se répercute pas sur les autres utilisateurs

Attribution de privilèges système

- Exemple:

```
GRANT CREATE SESSION,
      CREATE TABLE,
      EXECUTE ANY PROCEDURE
      TO iset;
```

Révocation de privilèges système

- REVOKE *priv._syst. / rôle* {, *priv._syst. / rôle*} FROM *user / rôle* {,*user/ rôle*} / PUBLIC

Vues du dictionnaire et privilèges système

- SYS.DBA_SYS_PRIVS: décrit tous les privilèges système attribués aux utilisateurs et aux rôles

```
select Grantee,Privilege
        from sys.dba_sys_privs
        where grantee in ('CONNECT','RESOURCE');

GRANTEE                        PRIVILEGE
------------------------------ ------------------------------
CONNECT                        ALTER SESSION
CONNECT                        CREATE CLUSTER
CONNECT                        CREATE DATABASE LINK
CONNECT                        CREATE SEQUENCE
CONNECT                        CREATE SESSION
CONNECT                        CREATE SYNONYM
CONNECT                        CREATE TABLE
CONNECT                        CREATE VIEW
RESOURCE                       CREATE CLUSTER
RESOURCE                       CREATE PROCEDURE
RESOURCE                       CREATE SEQUENCE
RESOURCE                       CREATE TABLE
RESOURCE                       CREATE TRIGGER
13 rows selected.
```

Les privilèges objet

- Un privilège objet permet d'exécuter une action particulière sur une table, vue, fonction, séquence, procédure, fonction ou package d'un schéma.

Privilèges Objet	Commentaires
SELECT,DELETE	
INSERT,UPDATE	spécifications de colonnes possibles
EXECUTE	procédure, fonction
ALTER	table, séquence
INDEX	table
REFERENCES	clé étrangère

Attribution de privilèges objet

- GRANT *priv._obj.*[(*liste_de_colonnes*)]
  {,*priv._obj.*[(*liste_de_colonnes*)]} / ALL
  ON [*schéma.*]*objet*
  TO *user / rôle* {,*user/ rôle*} / PUBLIC
  WITH GRANT OPTION
  - WITH GRANT OPTION permet aux bénéficiaires des privilèges de transmettre tout ou partie de ces privilèges à d'autres utilisateurs;
  - retirer des privilèges à un utilisateur qui les a reçus avec la clause WITH GRANT OPTION retire en cascade ces privilèges à tous les utilisateurs auxquels il les a transmis.

Attribution de privilèges objet

- Exemples:

  GRANT SELECT, UPDATE(adr,ville) ON fournisseurs TO PUBLIC;

  GRANT ALL ON entete TO joël, robert, marcel;

  GRANT EXECUTE ON conception TO eric, cecilia WITH GRANT OPTION;

Révocation de privilèges objet

- REVOKE *priv._obj.* {, *priv._syst.*} / ALL

  ON [*schèma.*]*objet*

  FROM *user/rôle* {,*user/rôle*} / PUBLIC

  [CASCADE CONSTRAINTS]

  - CASCADE CONSTRAINTS permet de supprimer toutes les contraintes d'intégrité référentielle définies sur les objets pour lesquels on demande le retrait de privilèges.

Vues du dictionnaire et privilèges objets

- USER_TAB_PRIVS [_MADE / _RECD] décrit les privilèges objets donnés ou reçus directement
- USER_COL_PRIVS [_MADE / _RECD] décrit les colonnes spécifiées dans les privilèges
- DBA _TAB_PRIVS décrit tous les privilèges sur tous les objets
- DBA _COL_PRIVS décrit toutes les colonnes spécifiées dans les privilèges

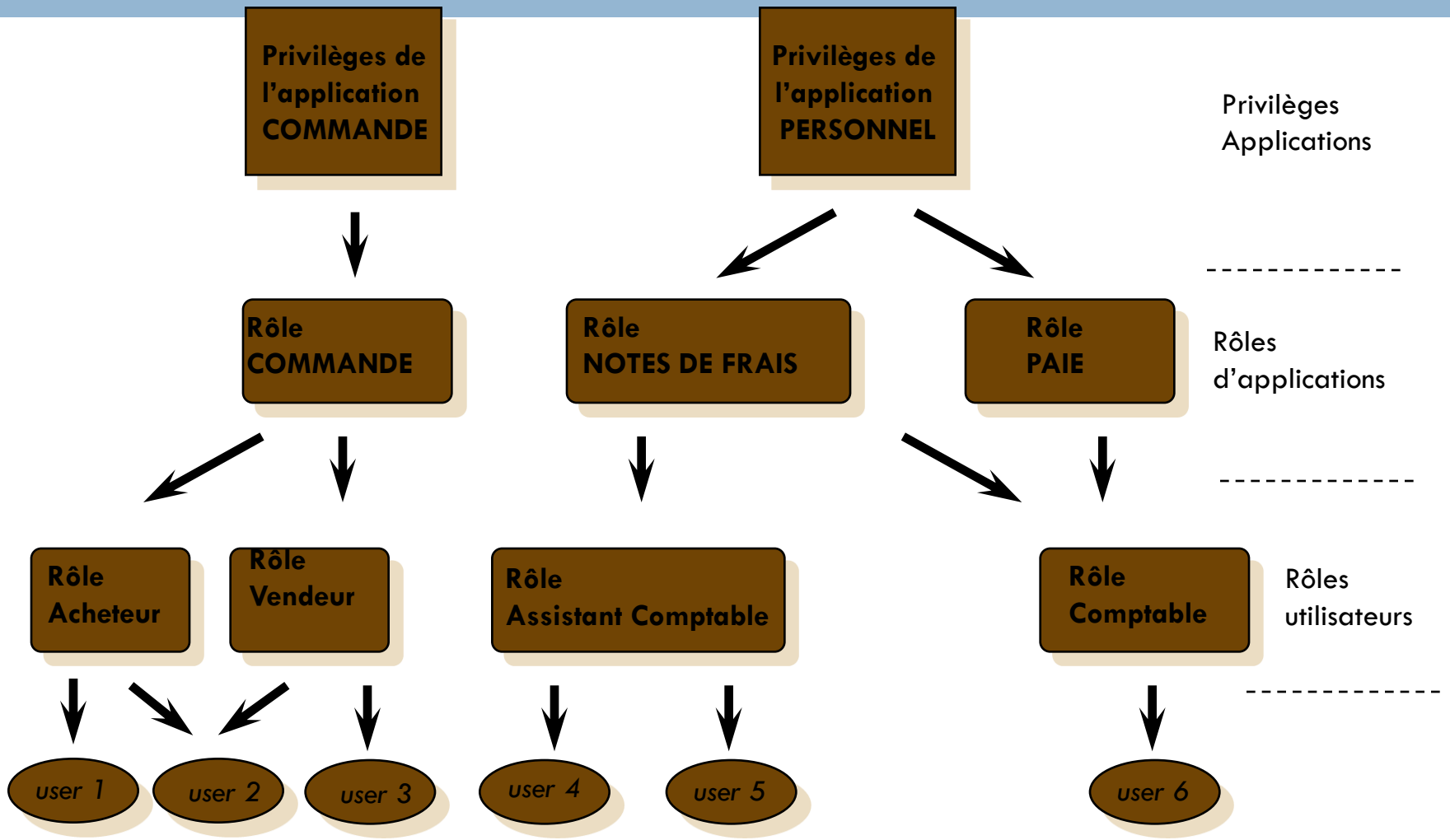
Vues du dictionnaire et objets

- ALL_TAB_PRIVS [_MADE / _RECD] décrit les privilèges objets donnés ou reçus directement, via un rôle ou PUBLIC
- ALL_TAB_PRIVS_MADE décrit les privilèges objets donnés via un rôle GRANT OPTION
- ALL_TAB_PRIVS_RECD décrit les privilèges objets reçus directement, via un rôle ou PUBLIC
- description sur les colonnes en remplaçant dans les noms de vues _TAB par _COL

Les rôles

- Ensemble nommé de privilèges:
  - composé de privilèges système ou/et objet
  - attribué à un utilisateur ou/et à un autre rôle
  - activé ou désactivé pour chaque utilisateur
  - pouvant nécessiter une autorisation
  - n'appartenant à aucun schéma
- Avantage:
  - centralise la gestion des privilèges
  - différencie les privilèges nécessaires aux applicatifs (rôle d'application) et ceux propres aux utilisateurs (rôle utilisateur)

Utilisation des rôles



Création, modification et suppression d'un rôle

- Création et modification d'un rôle:

CREATE / ALTER ROLE *rôle*

[NOT IDENTIFIED

/ IDENTIFIED BY *mot_de_passe* / EXTERNALLY]

- Suppression d'un rôle:

DROP ROLE *rôle*

- Exemple:

CREATE ROLE admin

IDENTIFIED BY istration

Activation d'un rôle

- SET ROLE *rôle* [IDENTIFIED BY *mot_de_passe*] {,*rôle* [IDENTIFIED BY *mot de passe*]}

  /ALL [EXCEPT *rôle* {, *rôle* }]

  /NONE

  - NONE: désactive tous les rôles de la session courante

- Exemple:

  SET ROLE appli_partie_1; .....

  SET ROLE appli_partie_2; .....

  SET ROLE ALL EXCEPT appli_partie_1 ;

Vues du dictionnaire et rôles

- SESSION_PRIVS: privilèges actifs de l'utilisateur courant
- SESSION_ROLES: rôles actifs de l'utilisateur ...
- USER_ROLE_PRIVS: rôles de l'utilisateur ...
- DBA_ROLES: tous les rôles
- DBA_SYS_PRIVS: privilèges système attribués aux utilisateurs et aux rôles
- ROLE_ROLE_PRIVS: informations sur les rôles attribués à d'autres rôles

Création d'un utilisateur de la base

- CREATE USER *utilisateur*

  IDENTIFIED BY *mot_de_passe*/EXTERNALLY

  [DEFAULT TABLESPACE *tablespace*]

  [TEMPORARY TABLESPACE *tablespace*]

  [PROFILE *profil*]

  {QUOTA *entier*/UNLIMITED ON *tablespace*}

- Exemple:

  CREATE USER joël IDENTIFIED BY jojo

  DEFAULT TABLESPACE users

  TEMPORARY TABLESPACE temp

  PROFILE developpeur_forms

  QUOTA 22M ON users QUOTA 700K ON user_data;

Modification d'un utilisateur

- ALTER USER *utilisateur*
  [IDENTIFIED BY *mot_de_passe*/EXTERNALLY]
  [DEFAULT TABLESPACE *tablespace*]
  [TEMPORARY TABLESPACE *tablespace*]
  [PROFILE *profil*]
  {QUOTA *entier*/UNLIMITED ON *tablespace*}
  [DEFAULT ROLE *rôle*{,*rôle*}
  /ALL[EXCEPT *rôle*] /NONE]
- Exemple:
  ALTER USER joël DEFAULT ROLE admin;

Suppression d'un utilisateur

- DROP USER *utilisateur* [CASCADE]
  - CASCADE: tous les objets du schéma associé sont supprimés.

Suppression d'une session utilisateur

- ALTER SYSTEM KILL SESSION 'sid,n°série'
  - sid: numéro de session utilisateur
  - n°série:numéro de série de la session

    sont contenus dans la vue V$SESSION
- Exemple:

```
SELECT sid,serial#,username
   FROM v$session WHERE username ='JOEL';
                12      97      JOEL
ALTER SYSTEM KILL SESSION '12,97'
```

Vues du dictionnaire et utilisateurs

- USER_USERS: permet d'obtenir des informations sur l'utilisateur courant
- ALL_USERS et DBA_USERS: permettent d'obtenir des informations sur tous les utilisateurs
- USER_TS_QUOTAS: donne les quotas par tablespace pour l'utilisateur courant
- DBA_TS_QUOTAS: donne les quotas par tablespace pour tous les utilisateurs