

TP N°1 Administration des BD

Nom Complet : El Hadji Ahmadou GUEYE

2. Organisation générale du dictionnaire

2.1. Vues relatives aux objets accessibles à tous les utilisateurs

1. schéma de la relation correspondant à DICT

```
SQL> DESCRIBE DICT;
```

Name	Null?	Type
TABLE_NAME		VARCHAR2(128)
COMMENTS		VARCHAR2(4000)

```
SQL> _
```

2. Visualisation du contenu complet du dictionnaire DICT

```
SQL > SELECT * FROM DICT;
```

3. Rôle et structure des tables

- **ALL_CATALOG** : Fournit une liste des objets auxquels on a accès ainsi que ceux que nous possédons.

```
SQL> DESC ALL_CATALOG
```

Name	Null?	Type
OWNER	NOT NULL	VARCHAR2(128)
TABLE_NAME	NOT NULL	VARCHAR2(128)
TABLE_TYPE		VARCHAR2(11)

- ☐ OWNER : Propriétaire de l'objet
- ☐ TABLE_NAME: Nom de l'objet
- ☐ TABLE_TYPE: Type de l'objet

- **ALL_USERS** : Fournit une liste de tous les utilisateurs de la base de données visible par le user courant.

```
SQL> DESC ALL_USERS;
```

Name	Null?	Type
USERNAME	NOT NULL	VARCHAR2(128)
USER_ID	NOT NULL	NUMBER
CREATED	NOT NULL	DATE
COMMON		VARCHAR2(3)
ORACLE_MAINTAINED		VARCHAR2(1)

- ☐ USERNAME : Nom de l'utilisateur
 - ☐ USER_ID : ID de l'utilisateur
 - ☐ CREATED : Date de création de l'utilisateur
 - ☐ COMMON : Indique s'il s'agit ou non d'un "common user" les valeurs possible sont "YES" ou "NO"
 - ☐ ORACLE_MAINTAINED : Indique si l'utilisateur a été créé et est maintenu par des scripts fournis par Oracle
- **ALL_TAB_COMMENTS** : Contient les commentaires sur les tables et vues visible par l'utilisateur courant.

```
SQL> DESC ALL_TAB_COMMENTS;
Name                               Null?    Type
-----
OWNER                               NOT NULL VARCHAR2(128)
TABLE_NAME                         NOT NULL VARCHAR2(128)
TABLE_TYPE                         VARCHAR2(11)
COMMENTS                           VARCHAR2(4000)
```

- ☐ OWNER : Propriétaire de l'objet
 - ☐ TABLE_NAME : Nom de l'objet
 - ☐ TABLE_TYPE : Type de l'objet
 - ☐ COMMENTS : Commentaire sur l'objet
- **ALL_COL_COMMENTS** : Contient les commentaires sur les colonnes des tables et vues auxquelles l'utilisateur a accès.

```
SQL> DESC ALL_COL_COMMENTS;
Name                               Null?    Type
-----
OWNER                               NOT NULL VARCHAR2(128)
TABLE_NAME                         NOT NULL VARCHAR2(128)
COLUMN_NAME                       NOT NULL VARCHAR2(128)
COMMENTS                           VARCHAR2(4000)
```

- ☐ OWNER : Propriétaire de l'objet
 - ☐ TABLE_NAME : Nom de l'objet
 - ☐ COLUMN_NAME : Nom de la colonne
 - ☐ COMMENTS : Commentaire portant sur la colonne
- **ALL_TAB_PRIVS** : Décrit tous les privilèges

```
SQL> DESC ALL_TAB_PRIVS
Name                               Null?    Type
-----
GRANTOR                            VARCHAR2(128)
GRANTEE                            VARCHAR2(128)
TABLE_SCHEMA                       VARCHAR2(128)
TABLE_NAME                         VARCHAR2(128)
PRIVILEGE                          VARCHAR2(40)
GRANTABLE                          VARCHAR2(3)
HIERARCHY                          VARCHAR2(3)
COMMON                             VARCHAR2(3)
TYPE                               VARCHAR2(24)
```

- ❑ GRANTOR : Nom de l'utilisateur qui a donné le privilège
 - ❑ GRANTEE : Nom de l'utilisateur ou du rôle auquel l'accès a été accordé
 - ❑ TABLE_SCHEMA : Schéma de l'objet
 - ❑ TABLE_NAME : Nom de l'objet
 - ❑ PRIVILEGE : Privilège sur l'objet
 - ❑ GRANTABLE : Indique si le privilège a été accordé avec l'option GRANT (Valeurs possible YES et NO)
 - ❑ HIERARCHY : Indique si le privilège a été accordé avec l'option HIERARCHY
 - ❑ COMMON : Indique comment la subvention a été accordée. Valeurs possible YES (utilisation de CONTAINER=ALL) ET NO
 - ❑ TYPE : Type de l'objet.
- ALL_TRIGGERS : Contient les déclencheurs sur les tables visibles par l'utilisateur actuel

```
SQL> DESC ALL_TRIGGERS
```

Name	Null?	Type
OWNER		VARCHAR2(128)
TRIGGER_NAME		VARCHAR2(128)
TRIGGER_TYPE		VARCHAR2(16)
TRIGGERING_EVENT		VARCHAR2(246)
TABLE_OWNER		VARCHAR2(128)
BASE_OBJECT_TYPE		VARCHAR2(18)
TABLE_NAME		VARCHAR2(128)
COLUMN_NAME		VARCHAR2(4000)
REFERENCING_NAMES		VARCHAR2(422)
WHEN_CLAUSE		VARCHAR2(4000)
STATUS		VARCHAR2(8)
DESCRIPTION		VARCHAR2(4000)
ACTION_TYPE		VARCHAR2(11)
TRIGGER_BODY		LONG
CROSSEDITION		VARCHAR2(7)
BEFORE_STATEMENT		VARCHAR2(3)
BEFORE_ROW		VARCHAR2(3)
AFTER_ROW		VARCHAR2(3)
AFTER_STATEMENT		VARCHAR2(3)
INSTEAD_OF_ROW		VARCHAR2(3)
FIRE_ONCE		VARCHAR2(3)
APPLY_SERVER_ONLY		VARCHAR2(3)

Descriptions de chaque attribut cf. [ici](#).

4. Rôle et structure de :

- ALL_CONSTRAINTS : décrit les définitions de contraintes sur les tables visibles par l'utilisateur courant.
 - Description :

```
SQL> DESC ALL_CONSTRAINTS;
```

Name	Null?	Type
OWNER		VARCHAR2(128)
CONSTRAINT_NAME	NOT NULL	VARCHAR2(128)
CONSTRAINT_TYPE		VARCHAR2(1)
TABLE_NAME	NOT NULL	VARCHAR2(128)
SEARCH_CONDITION		LONG
SEARCH_CONDITION_UC		VARCHAR2(4000)
R_OWNER		VARCHAR2(128)
R_CONSTRAINT_NAME		VARCHAR2(128)
DELETE_RULE		VARCHAR2(9)
STATUS		VARCHAR2(8)
DEFERRABLE		VARCHAR2(14)
DEFERRED		VARCHAR2(9)
VALIDATED		VARCHAR2(13)
GENERATED		VARCHAR2(14)
BAD		VARCHAR2(3)
RELY		VARCHAR2(4)
LAST_CHANGE		DATE
INDEX_OWNER		VARCHAR2(128)
INDEX_NAME		VARCHAR2(128)
INVALID		VARCHAR2(7)
VIEW_RELATED		VARCHAR2(14)
ORIGIN_CON_ID		NUMBER

- **ALL_CONS_COLUMNS** : Décrit les colonnes accessibles à l'utilisateur actuel et qui sont référencées dans les contraintes.

```
SQL> DESC ALL_CONS_COLUMNS;
```

Name	Null?	Type
OWNER	NOT NULL	VARCHAR2(128)
CONSTRAINT_NAME	NOT NULL	VARCHAR2(128)
TABLE_NAME	NOT NULL	VARCHAR2(128)
COLUMN_NAME		VARCHAR2(4000)
POSITION		NUMBER

Les types de contraintes reconnues par Oracle sont les suivant et sont déductible depuis les valeurs prise par la colonne **CONSTRAINT_TYPE** de la table **ALL_CONSTRAINTS**:

- **Contrainte CHECK** : permet de spécifier une condition sur chaque ligne d'une table.
- **Contrainte de clé primaire** : attribut ou combinaison d'attribut permettant d'identifier de manière unique chaque enregistrement.
- **Contrainte d'unicité** : attribut ou combinaison d'attribut ne pouvant pas prendre les mêmes valeurs pour tuples différents.
- **Contraintes d'intégrité référentielle** : elles fournissent un mécanisme pour garantir que les données sont conformes aux directives spécifiées par l'administrateur de la base de données (Clé étrangère, non nullité, ...).

Au niveau de la table **ALL_CONSTRAINTS**, nous avons l'attribut **OWNER** nous permettant d'identifier le propriétaire de la contrainte, l'attribut **CONSTRAINT_NAME** qui décrit le nom de la contrainte ainsi que l'attribut **CONSTRAINT_TYPE** qui décrit le type de la contrainte. Suivant les valeurs prises par le champ **CONSTRAINT_TYPE** combiné au contenu des champs **OWNER** et **CONSTRAINT_TYPE** nous avons suffisamment d'informations concernant la contrainte.

Valeurs prise par **CONSTRAINT_TYPE** :

- C : Contrainte check
- P : Contrainte de clé primaire
- U : Contrainte d'unicité
- R : Contrainte d'intégrité référentielle
- V : Avec une option check portant sur une view
- O : Avec un read only sur une view

5. Les différents type d'objet standard

Les différents types d'objets reconnus par oracle sont : les index, les clusters, les vues, les synonymes, les "database links", les séquences, les programmes stockés, les procédures, les packages, les triggers, etc.

La vue correspondante est **ALL_TYPES**.

6.

Nombre d'objets référencés dans **ALL_CATALOG** : comme le montre la capture suivante dans **ALL_CATALOG** il y est référencé 46057 objets.

```
SQL> SELECT COUNT(*) FROM ALL_CATALOG;  
  
COUNT(*)  
-----  
46057
```

La somme du nombre d'enregistrements des tables vaut : 46063 en effet il y'a une différence non négligeable. Cette différence est en effet expliquée par les vues concrètes.