

# Administration des bases de données

Jean-Yves Antoine

<http://www.info.univ-tours.fr/~antoine/>

# Administration des bases de données

I – Dictionnaire de données  
& Import / Export de données

# OBJECTIFS

## 1.1. NOTIONS

- 1.1.1. Dictionnaire de données
- 1.1.2. Importation / Exportation de données
- 1.1.3. Désactivation de contraintes



## 1.2. PRATIQUES

- 1.2.1. Connaître la structure des catalogues Oracle et MySQL et savoir l'utiliser dans des activités d'administration
- 1.2.2. Savoir utiliser les principales fonctionnalités de SQL\*Loader (SGBD Oracle)
- 1.2.3. Savoir activer et désactiver de contraintes de manière fine sous le SGBD ORACLE





## Catalogue ou dictionnaire

- Présent dans tout SGBD multi-bases et multi-utilisateurs : essentiel à l'administration de la base
- Base de données interne gérant tous les objets (tables et leurs contraintes, vues, utilisateurs, droits, index etc ...) connus du SGBD

## Structure

- Une ou plusieurs tables par type d'objet géré par le SGBD
- Une table/vue joue souvent le rôle de point d'entrée dans cette base de données d'administration : liste l'ensemble des éléments du dictionnaire.

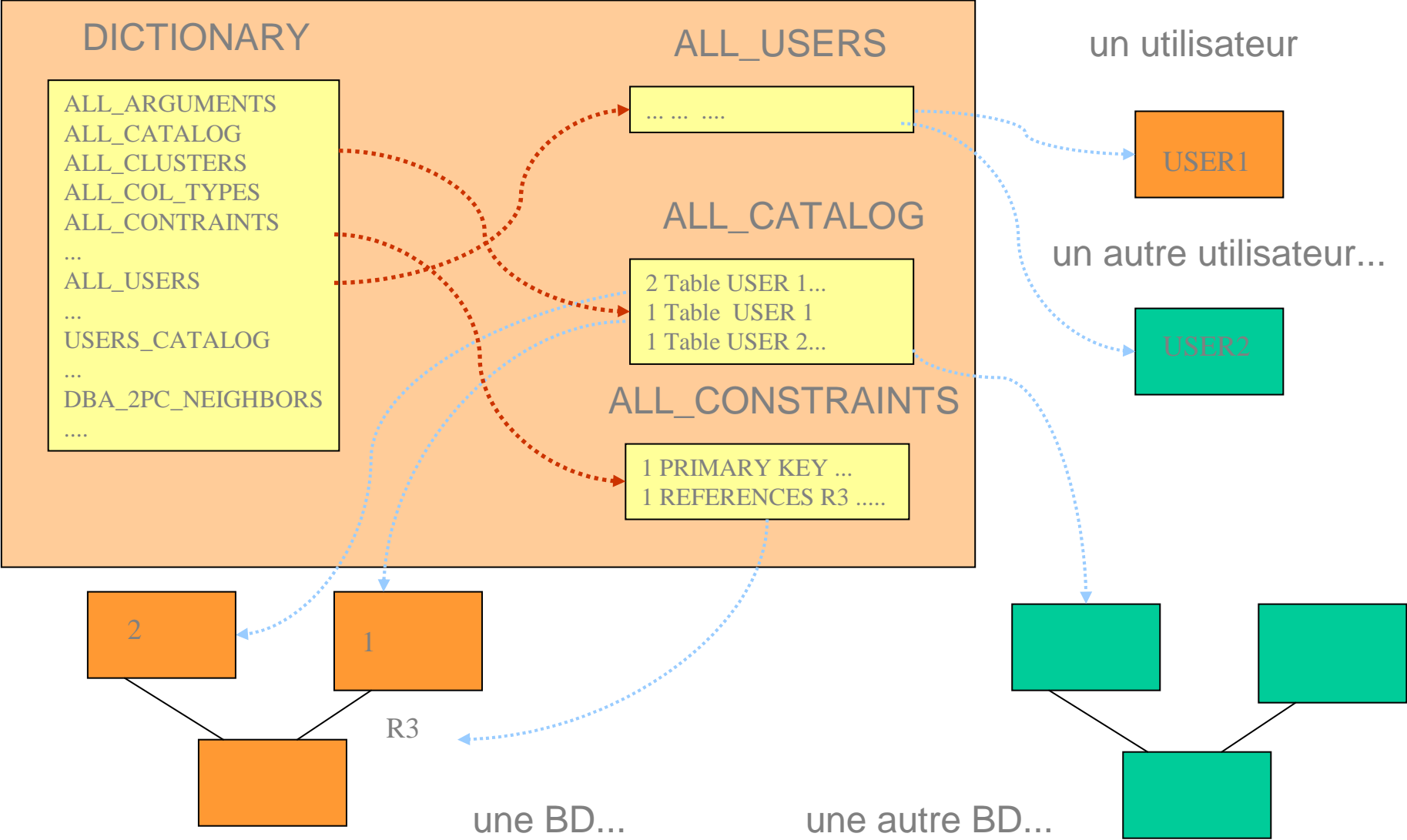
## Exemples

- **ORACLE**           vue **DICTIONARY**
- **MySQL**           vue **INFORMATION\_SCHEMA.TABLES** liste toutes les tables manipulées par le SGBD, dont celles du dictionnaire.

# DICTIONNAIRE DE DONNEES



## Exemple : dictionnaire ORACLE





## Interface conversationnelle SQL d'Oracle

- Implémente le standard SQL ANSI
- Intègre l'utilisation du langage procédural d'Oracle : PL/SQL
- Appel de l'interface conversationnelle :
  - ✓ commande `sqlplus` sous fenêtre de travail Oracle
  - ✓ Item menu `Application Development` puis `SQL+ Worksheet` dans l'environnement de travail `Enterprise Manager`

## Fichiers de commande SQL

- Fichier (extension `.sql`) comprenant des ordres SQL ou PL/SQL qui vont s'exécuter à l'appel du fichier
- Lancement à l'aide d'un ordre SQL\*Plus
  - `R(un) <nom_fichier_commande>`
  - icône éclair dans le `SQL+ Worksheet`



## Information cryptée accessible uniquement via des vues

- **USER\_\*** vues relatives aux objets appartenant à l'utilisateur
- **ALL\_\*** vues relatives aux objets accessibles par l'utilisateur (ceux dont il est propriétaire ainsi que ceux pour lesquels il a un droit d'accès)
- **DBA\_\*** vues relatives à l'administration (droits utilisateurs, *rollback segments*, etc...) de la BD. Accessibles uniquement par l'utilisateur SYSTEM.
- **V\$\*, GV\$\*** vues dynamiques relatives au suivi des performances

## Vue DICTIONARY

- Vue d'entrée du dictionnaire
- Raccourci : DICT



## Vues principales du dictionnaire

- **ALL\_CATALOG** Tous les objets (tables, vues,...) accessibles à l'utilisateur
- **USER\_CATALOG** Tous les objets qui sont propriétés de l'utilisateur
- **ALL\_COL\_COMMENTS** Commentaires sur le rôle des attributs des objets accessibles à l'utilisateur (donc listés dans ALL\_CATALOG)
- **ALL\_CONSTRAINTS** Toutes les contraintes d'intégrité (dont référentielles) sur les objets accessibles à l'utilisateur
- **ALL\_TAB\_PRIVS** Droits sur les objets accessibles par l'utilisateur. Ce sont les droits qui concernent l'utilisateur, celui qui a accordé les droits, le propriétaire ou le rôle PUBLIC (cf infra *GRANT*)
- **ALL\_USERS** Informations sur tous les utilisateurs de la base de données.
- **USER\_USERS** *A votre avis ?*





## Vues principales du dictionnaire

Une ou plusieurs vues par type d'objet manipulé par ORACLE

- Table
- Vue (`view`)
- Vue matérialisée (`materialized view`)
- Synonyme (`synonym`)
- Séquence (`sequence`)
- Cluster
- Index
- Déclencheur (`trigger`)

Objets «avancés »  
étudiés au cours  
de cet enseignement

## Exemple

- Vues `ALL_VIEWS`, `ALL_SYNONYMS` etc...



## Documentation BD ORACLE: commentaires stockés

- **ALL\_TAB\_COMMENTS**            Commentaires sur les tables/vues accessibles
- **ALL\_COL\_COMMENTS**        Commentaires sur les attributs des tables/vue

### Commenter un objet créé [SQL Oracle]

#### - Commentaire sur une table

```
COMMENT ON TABLE [<schema>].<nom_table>  
IS 'texte de commentaire' ;
```

#### - Commentaire sur un attribut

```
COMMENT ON COLUMN [<schema>].<tab>.<column>  
IS 'texte de commentaire' ;
```



## Importation à partir de fichiers de données

- Principe : données récupérées sous la forme d'un ensemble de fichiers ASCII / Unicode (données non structurées)
- Données externes ordonnées suivant un format donné

**Exemple**      format CSV : suite de champs de valeurs séparés par un point virgule

```
PT ; Parti des Travailleurs ; gauche
```

- Base de données à charger ayant sa propre structure interne
- ↳ Utilitaire assurant la lecture structurée des données et son chargement cohérent dans la base de données :
  - valeurs chargées dans les bons attributs de la base, conversion de type
  - ajout de valeur de clé primaire....

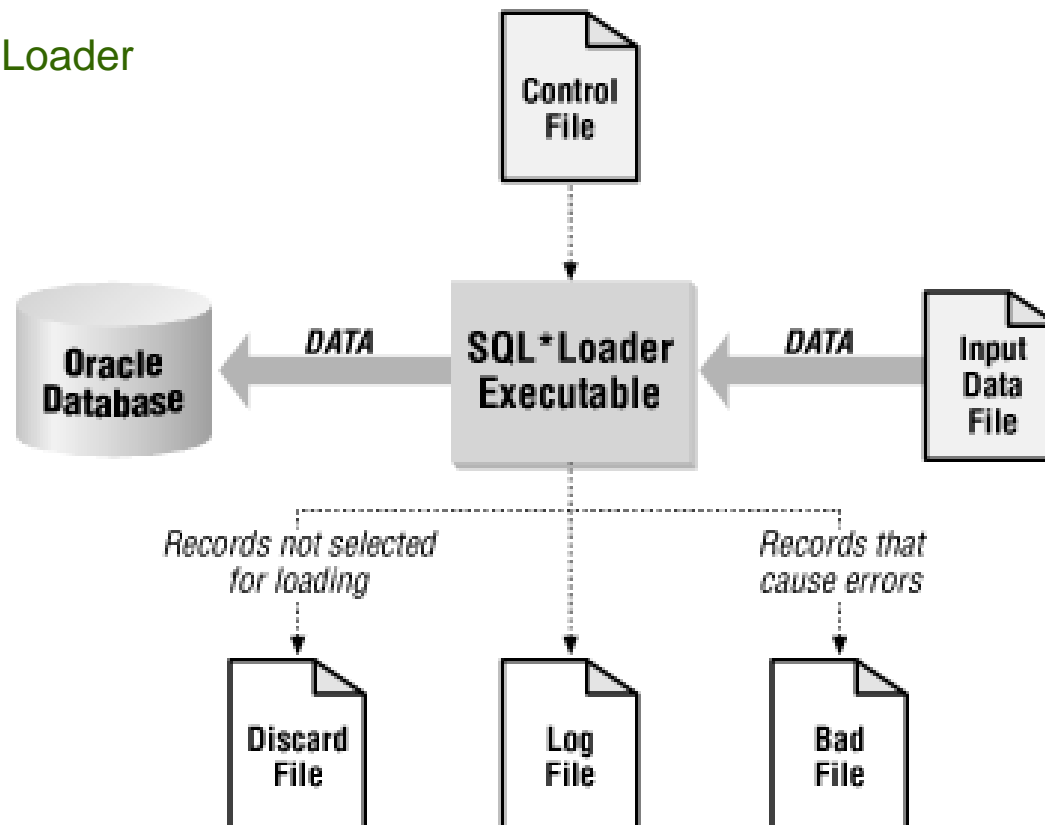
**Exemple : SQL\*Loader du SGBD ORACLE**



## Principe

Fichier de contrôle pour décrire le format des données à lire et le format de la base de données à charger

### Exemple ORACLE SQL\*Loader





## Exemple (suite) ORACLE SQL\*Loader

Fichier données

```
PS ; Parti Socialiste ; gauche ; 0
PC ; Parti Communiste ; gauche ; 0
Verts ; Parti Ecologiste ; gauche ; 0
Modem ; Mouvement Democrate ; centre ; 0
NC ; Nouveau Centre ; droite ; 1
UMP ; Union Majorite Presidentielle ; droite ; 1
FN ; Front National ; droite ; 0
```



Bases de Données

### RELATION

```
PARTI : [id]×[sigle]×[nom]×[courant]×[gvt]
```

### DEPENDANCES FONCTIONNELLES

```
[id] → [sigle]×[nom]×[courant]×[gvt]
```

### TYPE DE DONNES

```
[id], [gvt] : entier
```

```
[sigle], [nom], [courant] : chaîne caractères
```

### CONTRAINTES DOMAINE

```
[gvt] ∈ {0,1}
```



## Exemple (suite) ORACLE SQL\*Loader

Fichier de contrôle

```
LOAD DATA
INFILE 'parti.don'
INSERT
INTO TABLE parti
FIELDS TERMINATED BY ";"
(
    id          SEQUENCE "sequence_id.nextval",
    sigle       CHAR "upper(:sigle)" ,
    nom        CHAR ,
    courant    CHAR,
    gvt        INTEGER EXTERNAL
)
```

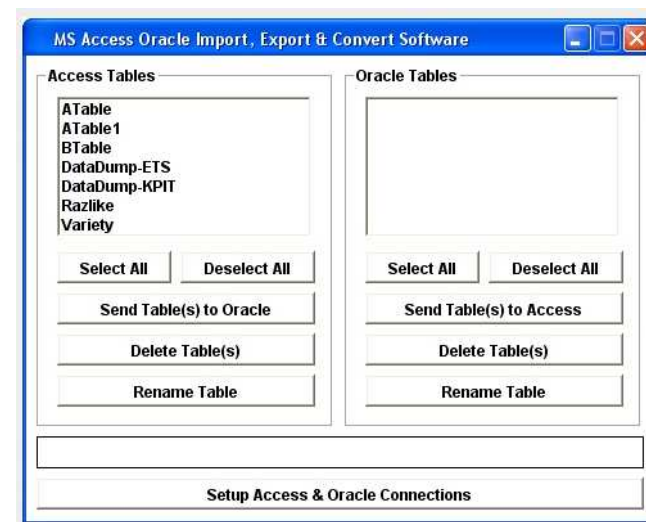


## Import / Export entre deux bases de données

- Principe : données structurées suivant le modèle relationnel
- Utilitaires spécifiques, limités ou pas au même SGBD

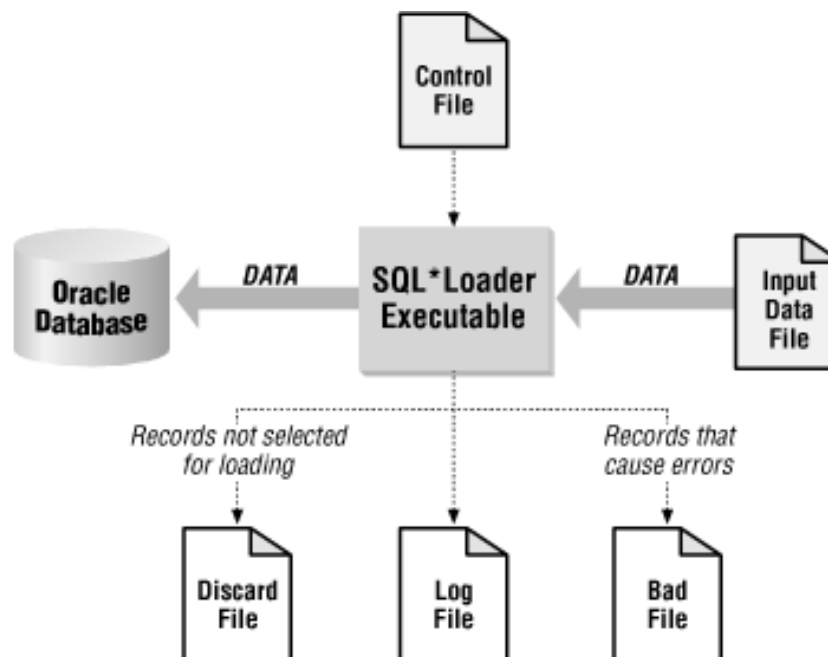
### Exemples

- ORACLE IMPORT et ORACLE EXPORT
- MySQL ORACLE IMPORT/EXPORT



## Exportation sous forme de fichier de données

- Format .csv le plus fréquent
- Format positionnel encore très utilisé (traitement par logiciels bancaires sous COBOL par exemple)



## Appel

- si le service est lancé, directement depuis l'interface de commande de l'OS.
- Commande simplifiée :

```
sqlldr [userid/passwd] nom_fic_ctl  
[nom_fic_data][nom_fic_log][nom_fic_bad]
```

## Aide en ligne

```
$ sqlldr
```





## Importance des fichiers de log

Trace d'exécution SQL\*Loader : pas d'information sur le chargement des données

```
$ sqlldr control=exemple log=exemple.log




SQL*Loader: Release 9.2.1.0.0 - Production on Wed Dec 5
2005
(c) Copyright 2001 Oracle Corporation. All rights reserved.

Commit point reached - logical record count 28
Commit point reached - logical record count 56
Commit point reached - logical record count 84
Commit point reached - logical record count 92
Commit point reached - logical record count 96
```



Problèmes non visibles : aller voir les fichiers .DISC et .BAD

## Localisation des données

- Données intégrées au fichier de contrôle 
- Localisation du fichier de données spécifiée dans le fichier de contrôle 
- Fichier de données spécifié dans la ligne de commande 

↳ Réutilisation du fichier de contrôle

## Segmentation du fichier de données par attribut

- Par leur position au sein de chaque ligne (colonnes de données alignées)
- A l'aide de séparateurs

## Exemple de fichier de données : parti.don

```
1 ; NC ; Nouveau Centre ; droite ; 1
2 ; LO ; Lutte Ouvrière ; ext_gauche ; 0
3 ; LCR ; Ligue Communiste Revolutionnaire ; ext_gauche ; 0
4 ; PS ; Parti Socialiste ; gauche ; 0
5 ; PC ; Parti Communiste ; gauche ; 0
6 ; PRG ; Parti Radical de Gauche ; gauche ; 0
7 ; Verts ; Parti Ecologiste ; gauche ; 0
8 ; Modem ; Mouvement Democrate ; centre ; 0
9 ; UMP ; Union Majorite Presidentielle ; droite ; 1
10 ; FN ; Front National ; ext_droite ; 0
11 ; CPNT ; Chasse Peche Nature Traditions ; droite ; 0
```



## Exemple de fichier de contrôle associé au fichier de données (1)

Chargement positionnel des données

insert, append, replace

```
LOAD DATA
INFILE 'parti.don'
INSERT
INTO TABLE parti
(
num_p POSITION(01:02) INTEGER EXTERNAL,
sigle POSITION(05:14) CHAR,
nom    POSITION(17:48) CHAR,
courant POSITION(52:62) CHAR,
gvt    POSITION(65:66) INTEGER EXTERNAL
)
```



## Exemple de fichier de contrôle associé au fichier de données (2)

Séparation des données à l'aide des séparateurs du format .csv

```
LOAD DATA
INFILDE 'parti.don'
INSERT INTO TABLE parti
FIELDS TERMINATED BY ";"
(
    num_p INTEGER EXTERNAL,
    sigle,
    nom,
    courant,
    gvt INTEGER EXTERNAL
)
```

terminated by / enclosed by

**TRAILING NULLCOLS**

(champs NULL  
en fin de ligne)

## Exemple de fichier de contrôle avec données intégrées

```
LOAD DATA
INSERT
INTO TABLE parti
FIELDS TERMINATED BY ";"
(
    num_p INTEGER EXTERNAL,
    sigle,
    nom,
    courant,
    gvt INTEGER EXTERNAL
)
BEGIN DATA
1 ; PT ; Parti des Travailleurs ; ext_gauche ; 0
2 ; LO ; Lutte Ouvrière ; ext_gauche ; 0
```

## Principaux formats externes reconnus par SQL\*Loader

SQL\*Loader assure la conversion du format externe au format interne à la base

Nom	Format
CHAR	Chaîne de caractères
DATE	Date sous forme de chaîne de caractères
INTEGER EXTERNAL	Entier sous forme de caractères
FLOAT EXTERNAL	Réel sous forme de caractères
DECIMAL EXTERNAL	Décimal sous forme de caractères
INTEGER	Entier sous forme <b>binaire</b>
DECIMAL	Décimal sous forme <b>binaire</b>
FLOAT	Réel sous forme <b>binaire</b>

Données ignorées : type FILLER      **exemple** FILLER INTEGER EXTERNAL

## Champs automatiques

- Séquences (cf. cours ch. III)      type SEQUENCE
- Date courante      type SYSDATE

## Modification des données avant chargement

Les fonctions SQL standard sont implantées dans SQL\*Loader

### Exemple

```
LOAD DATA
INFILE 'parti.don'
INSERT
INTO TABLE parti
FIELDS TERMINATED BY ";"
(
    pers_id          SEQUENCE "sequence_id.nextval",
    name             CHAR "upper(:name)" ,
    birthday        DATE "DD-MM-YYYY" ,
    fortune_fr      INTEGER EXTERNAL,
    fortune_euro    INTEGER EXTERNAL " :fortune_fr / 6.56"
)
```





## Chargement conditionnel des données

```
LOAD DATA
INFILE 'parti.don'
INSERT
INTO TABLE pers_fr
    WHEN pays='fr'
(
    pays    POSITION(01:02) CHAR,
    nom     POSITION(05:25) CHAR,
)
INTO TABLE pers_etranger
    WHEN pays !='fr'
(
    pays    POSITION(01:02) CHAR,
    nom     POSITION(05:25) CHAR,
)
```



## Présentation

- Outil de transfert du contenu d'une base vers une autre
- Sauvegardes partielles du contenu d'une base
- Ne concerne que des bases Oracle, éventuellement sur des plateformes différentes : non utilisable, comme SQL\*Loader, pour l'importation de données à partir de fichiers ASCII

## Lancement

Commande `exp nom_user/password`

## Oracle 10g et supérieurs

Outil complémentaire *Oracle Pump Export* annoncé comme plus rapide par Oracle



## Coût computationnel

- Opération lourde en terme de calculs
- Cette opération peut encore être ralentie lors du chargement dans la base de données
  - Vérification des contraintes
  - Remise à jour des index (cf. cours Patrick Marcel)
  - Vérification des conditions de déclenchement des triggers

## Recommandations

- Désactiver les contraintes avant le chargement et les réactiver ensuite
- Idem pour les index et les déclencheurs (triggers)

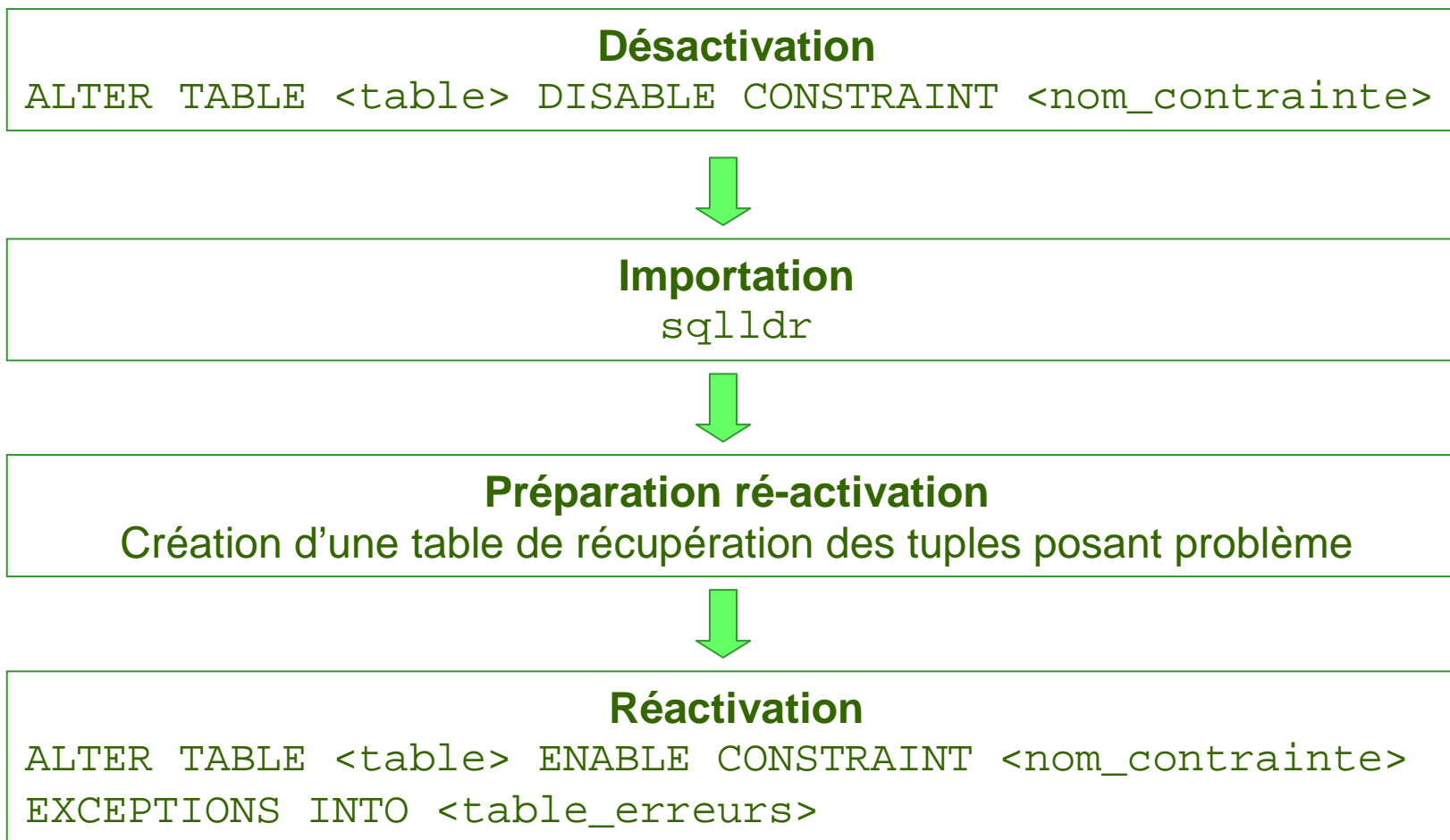


SQL : commande ALTER ...



## Désactivation de contraintes

Exemple : ORACLE





**Objectif** Accélérer les procédures d'importation / exportation

## Désactivation temporaire d'une contrainte

- Mise en état **DISABLE** de la contrainte

```
ALTER TABLE [<schema>.]<nom_table>  
DISABLE CONSTRAINT <nom_contrainte>
```

- Statut observable dans le dictionnaire : USER\_CONSTRAINTS

## Réactivation de la contrainte

- Mise en état **ENABLE** par un ALTER TABLE

```
ALTER TABLE [<schema>.]<nom_table>  
ENABLE CONSTRAINT <nom_contrainte>  
[EXCEPTIONS INTO <nom_table_erreur>];
```

- Liste des données violant la contrainte dans une table d'erreurs



## Table des erreurs récupérées

Table à créer spécifiquement à l'aide d'une commande CREATE TABLE

- 1° champ de type **ROWID** (contiendra les adresses dans la table des tuples érronés)
- 2° champ de type VARCHAR2(30) : nom du propriétaire de la table
- 3° champ de type VARCHAR2(30) : nom de la table
- 4° champ de type VARCHAR2(30) : nom de la contrainte violée

## Type ROWID

- Type prédéfini spécifique à ORACLE et correspondant à un pseudo-attribut qui précise l'adresse relative (dans la table) de chaque tuple. Qu'une relation ait ou pas une clé primaire, ORACLE lui associe un attribut ROWID (Row Identificateur).
- Lorsqu'un tuple enfreint une contrainte, celui-ci n'est pas effacé de la relation: simplement, son ROWID est marqué comme « non valide » et le tuple correspondant est ignoré lors des utilisations de la table.



## Consultation des erreurs à traiter après réactivation

- SELECT dans la table des erreurs
- Jointure avec la table concernée pour récupérer toutes les informations

```
CREATE TABLE Pbs_Load  
(tuple ROWID ...
```



```
SELECT ROWID, tab.* FROM table_chargée  
WHERE ROWID IN (select tuple FROM Pbs_Load);
```

## Validation de contrainte: modes VALIDATE / NO VALIDATE

Deux modes complémentaires :

- ENABLE / DISABLE vérification des contraintes lors des mises à jour
- (NO) VALIDATE vérification ou non pour les tuples déjà dans la table

### Exemple

```
ALTER TABLE [<schema>.]<nom_table>  
ENABLE NOVALIDATE CONSTRAINT <nom_contrainte>
```

↳ On ne vérifie pas les contraintes sur les données anciennes

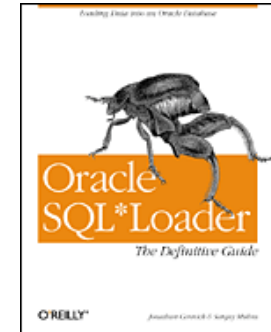
# BIBLIOGRAPHIE

## Ouvrages de références

GENNICK J., MISHRA S. (2004) *Oracle SQL\*Loader : the definite guide*. O'Reilly. ISBN 1-56592-948-9

BRIARD G. (2006) *Oracle 10g sous Windows*, Eyrolles, 2-212-11707-8

DELEGLISE D. (2004) *Guide du Développeur Oracle*, Supinfo, ISBN 2-914835-00-0



## Ouvrages disponibles à la B.U.

HEURTEL O. (2008) *Oracle 11i*, ENI, St Herblain, France (catalogue : 005.75 HEU)

HEURTEL O. (2005) *Oracle 10g*, ENI, St Herblain, France (005.75 HEU)

LONEY K., BRYLA B. (2005) *Oracle 10g, guide du DBA*. Oracle Press (005.75 LON)

MySQL AB (2005) *MySQL 5: Guide de l'administrateur*, Campus Pr., Paris (005.75 MYS)

NOIRAULT C. (2008) *Oracle 11g, entraînez-vous à administrer une base de données* ENI, St Herblain (005.75 NOI)

SOUTOU C. (2005) *SQL pour Oracle*, Eyrolles, ISBN 2-212-11697-7. (005.75 SOU)

THIBAULT C. (2005) *MySQL 5: Installation, mise en œuvre, administration, programmation*, ENI, St Herblain (005. THI)



# RESSOURCES ELECTRONIQUES

## Serveur de documentation Emery du département

- Oracle Database reference – Part. II The Static Data Dictionary
- Oracle Database Utilities – Part. II SQL\*Loader
  
- Référence MySQL 5 (en français) – Chap. 22 La base de données d'information INFORMATION\_SCHEMA