

Secteur Tertiaire Informatique
Filière étude - développement

Activité « Développer la persistance des données »

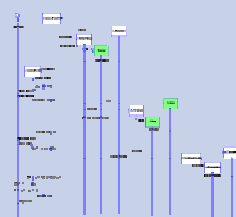
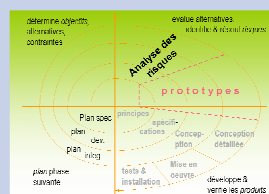
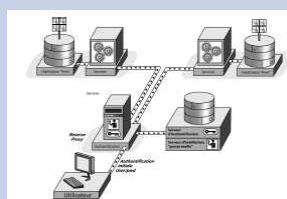
Mise en œuvre des transactions Oracle 11g

Accueil

Apprentissage

Période en entreprise

Evaluation



Code barre

SOMMAIRE

I	Transaction et verrous	4
I.1	Introduction	4
II	Définitions	5
III	Contrôle des transactions	5
III.1	Début de transaction	5
III.2	Fin d'une TRANSACTION	6
III.3	Découpage d'une transaction	6
III.4	Verrouillage des données	6
III.4.1	Concepts relatifs au verrouillage	6
III.4.2	Modes d'activation de verrouillage	7
III.4.3	Différents types de verrouillage	8
III.4.4	Verrouillage des données	8
III.4.5	Verrouillage du dictionnaire	11
III.4.6	Verrouillage interne	11

I TRANSACTION ET VERROUS

I.1 INTRODUCTION

L'un des objectifs d'un SGBD est de mettre à la disposition d'un grand nombre d'utilisateurs un ensemble cohérent de données. En présence de ce grand nombre d'utilisateurs, une attention particulière doit être apportée pour garantir la cohérence des données lors de leur manipulation simultanée par différents utilisateurs. Cette cohérence est assurée à l'aide des concepts de transaction et d'accès concurrents.

Toutes les modifications de données dans Oracle sont effectuées dans le cadre de transactions. Par défaut, ORACLE démarre une transaction pour chaque instruction individuelle et la valide automatiquement si l'exécution de l'instruction se termine normalement.

Une transaction est caractérisée les critères ACID

- **Atomique** : si une des instructions échoue, toute la transaction échoue
- **Cohérente**, car la base de données est dans un état cohérent avant et après la transaction, c'est-à-dire respectant les règles de structuration énoncées.
- **Isolée** : Les données sont verrouillées : il n'est pas possible depuis une autre transaction de visualiser les données en cours de modification dans une transaction.
- **Durable** : les modifications apportées à la base de données par une transaction sont validées

Par exemple, une transaction bancaire peut créditer un compte et en débiter un autre, ces actions devant être validées ensemble

II DEFINITIONS

Une transaction est une unité logique de traitements regroupant un ensemble d'opérations élémentaires (commandes SQL). Ces opérations doivent être soit exécutées entièrement, soit pas du tout, permettant ainsi à la base de données de passer d'un état cohérent à un nouvel état cohérent.

Exemple :

Soient les deux commandes suivantes :

```
INSERT INTO ligne_com  
VALUES (100, 1, 1001, 10);
```

et

```
UPDATE article  
SET qtestock = qtestock - 10  
WHERE idarticle = 1001;
```

La première commande enregistre une nouvelle ligne de commande et la seconde met à jour l'état du stock pour l'article commandé. Pour que la base de données reste dans un état cohérent, ces deux commandes doivent être exécutées et validées. Si, pour une raison quelconque, la seconde commande n'a pas pu être traitée, le système doit annuler la première.

Lorsque toutes les opérations constituant la transaction sont exécutées et deviennent effectives, nous disons que la transaction est validée. Les modifications apportées par les opérations élémentaires deviennent alors définitives. Dans le cas contraire où au moins une opération n'a pas pu être exécutée pour une raison quelconque (condition non vérifiée, accès impossible aux données, panne), nous disons que la transaction est annulée. Les modifications apportées par toutes les opérations élémentaires sont alors annulées et on revient à l'état qu'avait la base avant la transaction.

Dans le but d'assurer une souplesse dans la gestion des transactions, il est possible de découper une longue transaction en sous-transactions à l'aide de repères permettant d'effectuer, en cas de besoin, des annulations partielles de la transaction.

Cette notion de sous-transaction permet d'effectuer un découpage plus fin des unités de traitements.

III CONTROLE DES TRANSACTIONS

Le contrôle des transactions consiste à définir le début et la fin d'une transaction (validation ou annulation) ainsi que son découpage éventuel en sous-transactions.

III.1 DEBUT DE TRANSACTION

Il n'existe pas de commande permettant de marquer explicitement le début d'une transaction. Ce début est implicitement défini par la première commande SQL exécutée ou par la fin d'une transaction précédente. Le début d'une application ou d'une session SQL constitue automatiquement le début d'une transaction. De même,

la fin d'une transaction (par validation ou par annulation) marque le début d'une nouvelle transaction.

III.2 FIN D'UNE TRANSACTION

La fin d'une transaction peut être définie explicitement ou implicitement.

La fin explicite d'une transaction est réalisée à l'aide des deux commandes COMMIT ou ROLLBACK. La première valide les opérations élémentaires de la transaction et la seconde les annule.

Les événements suivants constituent une fin implicite d'une transaction :

- Exécution d'une commande de définition de données (CREATE, ALTER, RENAME et DROP) : toutes les opérations exécutées depuis le début de la transaction sont validées.
- fin normale d'un programme ou d'une session avec déconnexion d'Oracle : la transaction est validée.
- fin anormale d'un programme ou d'une session (sortie sans déconnexion d'Oracle) : la transaction est annulée.

III.3 DECOUPAGE D'UNE TRANSACTION

Il est possible dans Oracle, d'effectuer un découpage d'une transaction en insérant des points de repère (en anglais savepoints). A l'aide de ces points de repère il est possible d'annuler un sous-ensemble d'opérations d'une transaction à partir d'un point de repère.

La création d'un point de repère se fait à l'aide de la commande suivante :

```
SAVEPOINT point_repère;
```

Pour annuler les opérations à partir d'un point de repère, on utilise la ROLLBACK en précisant le point de repère :

```
ROLLBACK TO [SAVEPOINT] point_repère;
```

III.4 VERROUILLAGE DES DONNEES

III.4.1 Concepts relatifs au verrouillage

Pour que l'exécution simultanée de plusieurs transactions donne le même résultat qu'une exécution séquentielle, la solution utilisée consiste à verrouiller momentanément les données utilisées par une transaction jusqu'à la fin de la mise à jour. Les autres transactions demandant ces données sont mises en attente jusqu'à leur libération (déverrouillage). Cette technique de verrouillage permet d'éviter les interactions destructives des transactions, c'est à dire les opérations n'assurant pas l'intégrité des données.

Le verrouillage s'applique d'une façon générale à une Ressource. Une ressource peut correspondre aux objets créés par les utilisateurs tels que les tables (ou uniquement quelques lignes d'une table), ainsi qu'aux objets système tels que des éléments du dictionnaire ou des zones de données en mémoire centrale (SGA).

III.4.2 Modes d'activation de verrouillage

Le verrouillage des données peut être activé explicitement par l'utilisateur ou une application ou implicitement suite à l'exécution d'une commande de définition ou de manipulation de données.

III.4.2.1 Verrouillage implicite

Le SGBD Oracle effectue automatiquement tous les verrouillages nécessaires pour le maintien de la cohérence des données. Ces verrouillages sont effectués sans aucune intervention de l'utilisateur.

Par exemple, si plusieurs utilisateurs font UPDATE sur une même ligne sans valider, le deuxième sera en attente.

III.4.2.2 Verrouillage explicite

Dans certains cas, l'utilisateur peut ressentir le besoin de contrôler lui-même les mécanismes de verrouillage des données. Ce contrôle peut se faire au niveau des transactions ou au niveau d'une instance.

Au niveau de la transaction, le verrouillage explicite peut être activé dans les cas suivants :

1) En utilisant la commande :

```
LOCK TABLE nom_table IN mode_verrouillage MODE;
```

avec :

nom_table nom de la table à verrouiller.

mode_verrouillage mode selon lequel la table sera verrouillée. Il peut être : ROW SHARE, SHARE, ROW EXCLUSIVE, SHARE ROW EXCLUSIVE, et EXCLUSIVE.

La description de chacun de ces modes est donnée dans la suite.

La table nom_table est ainsi verrouillée selon le mode mode_verrouillage.

2) En utilisant la commande :

```
SELECT liste_selection  
FROM nom_table  
WHERE condition  
FOR UPDATE
```

Les lignes de la table nom_table vérifiant la condition 'condition' sont verrouillées.

3) La consistante de lecture (read consistency) est appliquée par défaut à toutes les transactions. Elle a pour conséquence de rendre visible à une transaction T2 toute modification apportée aux données (et validée) par une transaction T1, même si ces modifications interviennent pendant le déroulement de la transaction T2. Dans certains cas, on ne souhaite pas voir les modifications qui ont lieu après le début d'une transaction. Ce mode est appelé READ ONLY. Pour passer du mode par défaut au mode READ ONLY on utilise la commande suivante au début d'une transaction :

```
SET TRANSACTION READ ONLY;
```

Ce mode:

- Interdit tout Update, Insert, Delete dans la transaction
- Ne rend plus visible les modifications

Ce mode reste valable jusqu'à la fin de la transaction. Le mode par défaut est ensuite rétabli.

Remarque : dans tous les cas, si une mise à jour n'a pas été validée, elle n'est pas visible par les autres utilisateurs.

III.4.3 Différents types de verrouillage

La technique de verrouillage peut être appliquée à trois types de ressources : les données, le dictionnaire de données et la mémoire centrale. A chacun de ces ressources correspond un type de verrouillage.

III.4.4 Verrouillage des données

Le verrouillage des données, appelé aussi verrouillage du langage de manipulation de données permet de maintenir la cohérence des données en cas d'accès concurrents. Oracle utilise deux niveaux de verrouillage des données : verrouillage des lignes (row locks) et verrouillage des tables. La combinaison de ces deux niveaux de verrouillage donne lieu aux cinq verrouillages suivants :

- mode lignes partagées (Row Share ou RS),
- mode lignes exclusives (Row exclusive ou RX),
- mode table partagée (Share ou S),
- mode partage exclusif de lignes (Share row exclusive ou SRX),
- mode table exclusive (Exclusive ou X).

Le verrouillage des données prend fin après la validation ou l'annulation d'une transaction. Lorsqu'une transaction est découpée en sous-transactions, les données verrouillées à partir d'un point de repère sont libérées dès qu'une annulation partielle de la transaction est effectuée pour ce point de repère.

III.4.4.1 Verrouillage en mode lignes partagées (Row Share ou RS) :

Ce mode de verrouillage entraîne un verrouillage sélectif des lignes en vue de leur modification. Lorsqu'une table est verrouillée en mode lignes partagées, toutes les opérations et tous les autres modes de verrouillage sont autorisés, à l'exception du verrouillage exclusif de la table (X).

Ce mode peut être activé :

- Explicitement à l'aide de la commande :

```
LOCK TABLE nom_table IN ROW SHARE MODE;
```

- Ou implicitement lors de l'exécution de la commande de sélection suivante :

```
SELECT ...  
FROM nom_table  
WHERE...  
FOR UPDATE ;
```

Dans ce cas, on ne peut pas faire ensuite Update ou Select for Update sur le même ligne.

Ce mode est le moins restrictif et permet donc un haut niveau d'accès concurrents.

III.4.4.2 Verrouillage en mode lignes exclusives (Row exclusive ou RX) :

Ce mode de verrouillage entraîne aussi un verrouillage sélectif de lignes d'une table lors de leur modification. Ces lignes ne peuvent pas être partagées par d'autres transactions. Lorsqu'une table est verrouillée en mode lignes exclusives, toutes les opérations et tous les autres modes de verrouillage sont autorisés, à l'exception des modes suivants : table partagée (S), lignes partagées (SR) et table exclusive (X)

Ce mode peut être activé :

- Explicitement l'aide de la commande :

```
LOCK TABLE nom_table IN ROW EXCLUSIVE MODE;
```

- Ou implicitement lors de l'exécution de l'une des commandes suivantes :

```
INSERT INTO nom_table ...,  
UPDATE nom_table ...;  
DELETE FROM nom_table ...;
```

Ce mode est plus restrictif que le mode RS.

III.4.4.3 Verrouillage en mode table partagée (Share ou S) :

Ce mode permet de verrouiller entièrement une table tout en permettant aux autres transactions d'effectuer sur cette table des sélections, des verrouillages en mode lignes partagées (RS) ou en mode table partagée (S). Il est donc possible d'avoir plusieurs transactions qui effectuent ce mode de verrouillage simultanément sur la même table. Toutes les opérations modifiant le contenu de la table (insertion, suppression et mise à jour) ne peuvent pas être exécutées. Il n'est pas possible non plus de verrouiller cette table en mode partage exclusif de lignes (SRX), ni en mode table exclusive (X).

Lorsque plusieurs transactions verrouillent simultanément une même table en mode table partagée (S), aucune de ces transactions ne peut effectuer une mise à jour sur cette table. Par contre, une transaction ayant verrouillé une table en mode table partagée peut la mettre à jour (insertion, suppression ou mise à jour) si aucune autre transaction ne l'a verrouillée selon ce mode.

Ce mode ne peut être activé qu'explicitement à l'aide de la commande :

- LOCK TABLE nom table IN SHARE MODE;

III.4.4.4 Verrouillage en mode partage exclusif de lignes (Share row exclusive ou SRX):

Avec ce mode de verrouillage, les autres transactions ne peuvent effectuer que des opérations de sélection ou de verrouillage en mode lignes partagées (SR). Une seule transaction peut effectuer ce mode de verrouillage sur une table. Toutes les opérations modifiant le contenu de la table (insertion, suppression ou mise à jour) ne peuvent pas être exécutées. Il n'est pas possible non plus de verrouiller cette table selon les modes suivants : table partagée (S), partage exclusif de lignes (SRX) ou table exclusive (X).

Ce mode ne peut être activé qu'explicitement à l'aide de la commande :

- LOCK TABLE nom table IN SHARE ROW EXCLUSIVE MODE;

Ce mode est plus restrictif que le mode table partagée.

III.4.4.5 Verrouillage en mode table exclusive (Exclusive ou X) :

Ce mode de verrouillage permet à la transaction qui le demande d'effectuer une écriture exclusive sur la table. Les autres transactions ne peuvent effectuer que des opérations de sélection sur cette table. Une seule transaction peut effectuer ce mode de verrouillage sur une table. Toutes les opérations modifiant le contenu de la table (insertion, suppression ou mise à jour) ne peuvent pas être exécutées. Il n'est pas possible non plus d'effectuer aucun mode de verrouillage.

Ce mode de verrouillage est le plus restrictif. Il ne peut être activé qu'explicitement à la commande :

- LOCK TABLE nom table IN EXCLUSIVE MODE;

III.4.4.6 Compatibilité entre les modes de verrouillage :

Le tableau suivant donne des compatibilités entre les différents modes de verrouillage

Mode de Verrouillage	Commandes SQL correspondantes	Compatibilité avec les autres modes de verrouillage				
		RS	RX	S	SR X	X
RS : Lignes partagées	Select ... from ... for update Lock table ... in row share mode	O	O/N	O	O	N
RX : lignes exclusives	Insert into ... Update ... Delete from ... Lock table ... in row exclusive mode	O/N	O/N	N	N	N
S : Table partagée	Lock table ... in share mode	O	N	O	N	N
SRX : partage exclusif de lignes	Lock table... in share exclusive mode	O				
X :table exclusive	Lock table... in exclusive mode	N	N	N	N	N

III.4.5 Verrouillage du dictionnaire

Le verrouillage du dictionnaire des données, appelé aussi verrouillage du langage de définition de données, permet de protéger la définition des objets de la base (tables, vues, utilisateurs, procédures, etc...) lorsque ces objets sont manipulés par des commandes de définition de données dans une transaction. Par exemple, lors de la création d'une vue, Oracle verrouille la définition des tables référencées par cette vue, interdisant ainsi toute modification de structure ou suppression des tables.

Le verrouillage du dictionnaire de données est effectué automatiquement par les différentes commandes du langage de définition des données. Il n'existe donc pas de commandes explicites de verrouillage analogues à celles du verrouillage des données.

Les objets du dictionnaire de données peuvent être verrouillés selon deux modes :

- Verrouillage exclusif : il s'agit de verrouiller certaines ressources en mode exclusif de façon à interdire leur utilisation par toute commande de définition ou de manipulation de données. C'est le cas de la plupart des commandes de définition de données. Par exemple, lors de la suppression ou de la modification de structure d'une table, un verrouillage exclusif est nécessaire afin d'interdire toute utilisation de cette table pendant cette opération.
- Verrouillage partagé : certaines commandes de définition de données ne nécessitent pas le verrouillage exclusif de la ressource correspondant. Dans ce cas, un verrouillage partagé est suffisant. C'est le cas, par exemple, lors de la création d'une vue. Un verrouillage partagé est suffisant sur les tables de la base. Ce type de verrouillage est utilisé pour les commandes de création des tables, des vues, des procédures, des synonymes et des packages.

Etant donné que chaque commande de définition de données met fin implicitement à une transaction, le(s) verrouillage(s) correspondant(s) au niveau du dictionnaire de données prennent fin juste après l'exécution de la commande.

III.4.6 Verrouillage interne

Ce troisième type de verrouillage concerne la mémoire utilisée par Oracle pour l'exécution du noyau et des applications. Il est complètement transparent aux utilisateurs. Il existe un type particulier de verrous internes appelés loquets (Latches).

Ce sont des mécanismes internes d'Oracle permettant la protection des structures partagées de la SGA telles que la liste des utilisateurs actuels de la base. Ce type de verrouillage est activé et désactivé par les différents processus internes d'Oracle (processus serveurs ou d'arrière-plan).

Etablissement référent
Direction de l'ingénierie Neuilly

Equipe de conception
Groupe d'étude de la filière étude - développement

Remerciements :

Reproduction interdite

Article L 122-4 du code de la propriété intellectuelle.
« toute représentation ou reproduction intégrale ou partielle faite sans le
consentement de l'auteur ou de ses ayants droits ou ayants cause est
illicite. Il en est de même pour la traduction, l'adaptation ou la reproduction
par un art ou un procédé quelconques. »

Date de mise à jour 20/06/2008
afpa © Date de dépôt légal juin 08



afpa / Direction de l'Ingénierie 13 place du Générale de Gaulle / 93108 Montreuil
Cedex
association nationale pour la formation professionnelle des
adultes
Ministère des Affaires sociales du Travail et de la
Solidarité