

Mise en œuvre d’une application de gestion de contrôle d’accès

Analyse Transactionnelle

**Nicolas Jérémy**

**Groupe 3 Vaschalde Rudy**

**Michel Rémi**

**Quinard Cyril**

2013

****

Sommaire

[I - Identification des transactions 3](#_Toc346704274)

[A. Les transactions liées à l’administrateur 3](#_Toc346704275)

[a. Gestion des bâtiments 3](#_Toc346704276)

[b. Gestion des groupes de bâtiments 4](#_Toc346704277)

[c. Gestion des points d’accès 5](#_Toc346704278)

[d. Gestion des jours fériés 5](#_Toc346704279)

[B. Les transactions liées au superviseur 5](#_Toc346704280)

[a. Gestion des personnes et des badges 5](#_Toc346704281)

[b. Gestion des plages d’accès d’un groupe de bâtiment 6](#_Toc346704282)

[c. Gestion des plages d’accès d’un groupe de personnes 6](#_Toc346704283)

[d. Gestions des réservations de salles 6](#_Toc346704284)

[C. Les transactions liées à l’utilisateur 6](#_Toc346704285)

[a. Passage d’un point d’accès 6](#_Toc346704286)

# Identification des transactions

## Les transactions liées à l’administrateur

### Gestion des bâtiments

* + Ajout d’un bâtiment: **READ COMMITTED** 
    - Insérer le bâtiment
  + Suppression d’un bâtiment: **SERIALIZABLE**
    - Supprimer toutes les réservations du bâtiment
    - Supprimer toutes les salles du bâtiment
    - Supprimer le bâtiment

Lors de la suppression d’un bâtiment, un trigger est lancé pour supprimer les salles de ce bâtiment ainsi que les réservations correspondantes. Ces opérations sont critiques et nécessite d’être réalisées en mode SERIALIZABLE (voir exemple ci-dessous).  
  
Exemple en mode READ COMMITTED (posant problème):

|  |  |
| --- | --- |
| **Transaction 1** | **Transaction 2** |
| Suppression de toutes réservations pour les salles du bâtiment |  |
|  | Insertion d’une réservation sur une des salles du bâtiment |
|  | Validation |
| Suppression des salles du bâtiment → problème car T2 a rajouté une référence dans réservation après que T1 les ait supprimées |  |
| Suppression du bâtiment |  |
| Validation |  |

* + Modification : **READ COMMITTED**

(Réaffectation du bâtiment ou changement d’adresse)

* + - Modifier groupe de bâtiment (ou adresse) dans Batiment

### Gestion des groupes de bâtiments

* + Ajout: **READ COMMITTED - non DEFERRABLE**
    - Ajouter groupe\_bat
    - Ajouter meta-groupe

Lorsqu’un superviseur souhaite ajouter un groupe de bâtiment, nous allons, au niveau applicatif, réaliser une transaction qui va ajouter le groupe bâtiment dans la table GROUPE\_BAT ainsi que son meta-groupe correspondant dans la table GROUPE. Ces opérations ne présentent pas de réel problème de concurrence, c’est pourquoi nous avons décidé de réaliser cette transaction en mode READ COMMITTED. La cohérence étant toujours garantie au cours de l’exécution de la transaction, il n’est pas nécessaire d’évaluer les contraintes en mode différé.

|  |  |
| --- | --- |
| Transaction T1 | Transaction T2 |
| Ajout d’un groupe de bâtiment |  |
|  | Ajout d’un groupe de bâtiment |
|  | Ajout du meta-groupe |
|  | Validation |
| Ajout du meta-groupe |  |
| Validation |  |

* + Suppression: **SERIALIZABLE**
    - Supprimer les réservations
    - Supprimer les autorisations correspondant au groupe de bâtiment
    - Mettre les bâtiments du groupe de bâtiment à NULL
    - Supprimer le meta-groupe
    - Supprimer le groupe\_bat

|  |  |
| --- | --- |
| **Transaction 1** | **Transaction 2** |
|  | Suppression du groupe de bâtiment n°2 |
| Modification du bâtiment vers le groupe de bâtiment n°2 |  |
| Validation |  |
|  | Validation |

Au niveau applicatif, forcer le superviseur à réaffecter les bâtiments dont le groupe de bâtiment vient de passer à NULL (Cas d’utilisation Modification d’un bâtiment)

### Gestion des points d’accès

### Gestion des jours fériés

## Les transactions liées au superviseur

### Gestion des personnes et des badges

* Ajouter personne: **READ COMMITTED**
* Placer l’évaluation des contraintes en mode DEFERRABLE (Respect de la contrainte: Une personne doit avoir au moins 1 badge actif)
* Insérer la nouvelle personne
* Insérer un nouveau badge
* Lier la nouvelle personne et le nouveau badge

*Scénario:* 2 transactions concurrentes ajoutent une personne avec le même et le même prénom.

|  |  |
| --- | --- |
| Transaction T1 | Transaction T2 |
|  | Ajout d’une personne ayant pour nom et prénom (nom1, prenom1) |
| Ajout d’une personne ayant pour nom et prénom (nom1, prenom1) |  |
|  | Ajout d’un badge étant en état actif |
|  | Ajout d’une affectation entre personne et badge |
| Ajout d’un badge étant en état actif |  |
|  | Validation |
| Ajout d’une affectation entre personne et badge |  |
| Validation |  |

**Remarque : 2 personnes physiques peuvent avoir les mêmes nom/prénom. La clé porte sur l’identifiant de la personne**

* Modifier un badge : **READ COMMITED**

* Supprimer une personne: **SERIALIZABLE**

- Placer l’évaluation des contraintes en mode DEFERRABLE (Respect de la contrainte: Une personne doit avoir au moins 1 badge actif)

- Rechercher le badge affecté à la personne supprimée

- Changer l’état du badge Actif → Inactif

- Supprimer toutes les références de la personne (Affectation, Membre, Accès

- Supprimer la personne dans PERSONNE

*Scénario:* 1 transaction tente de supprimer une personne alors qu’une autre transaction tente d’affecter cette personne à un groupe

Exemple en mode **READ COMMITTED** (posant problème):

|  |  |
| --- | --- |
| Transaction T1 | Transaction T2 |
|  | Sélection des personnes à supprimer dans la table affectation |
|  | Mise à jour de la table badge mettant le badge de la personne en inactif |
|  | Suppression des références de la personne dans les tables affectation, membre et accès |
| Affectation de la personne à un groupe  → problème car T1 affecte une personne à un groupe ayant été supprimé par T2 |  |
| VALIDATION |  |
|  | Suppression de la personne |
|  | VALIDATION |

### Gestion des plages d’accès d’un groupe de bâtiment

### Gestion des plages d’accès d’un groupe de personnes

### Gestions des réservations de salles

## Les transactions liées à l’utilisateur

### Passage d’un point d’accès