



TP mémoire transactionnelle

Badr SAJID

Département Sciences du Numérique - Première année
2020-2021

Table des matières

1	Thèmes	3
2	Préparation	3
3	Exemple et scénario d'exécution	4
3.1	Scénario 1	4
3.1.1	Politique de contrôle de concurrence pessimiste sans blocage, avec propagation directe : TMPP	4
3.1.2	politique de contrôle de concurrence par certification avec propagation différée : TMPC	4
3.1.3	réalisant le protocole de verrouillage à 2 phases avec propagation directe : TM2PL	5
3.2	Scénario 2	6
3.2.1	Politique de contrôle de concurrence pessimiste sans blocage, avec propagation directe : TMPP	6
3.2.2	politique de contrôle de concurrence par certification avec propagation différée : TMPC	6
3.2.3	réalisant le protocole de verrouillage à 2 phases avec propagation directe : TM2PL	6
4	Évaluation de protocoles	8

1 Thèmes

- Exemples et scénarios de programmation concurrente transactionnelle.
- Problèmes de cohérence induits par la mémoire transactionnelle.
- Mise en œuvre de différents protocoles de propagation et de contrôle de concurrence.
- Évaluation des performances et de l'intérêt de différents protocoles de contrôle de concurrence, selon le contexte d'exécution et le profil des applications.

2 Préparation

Le TP propose un service de mémoire transactionnelle, permettant d'implanter différentes politiques de propagation des écritures et de contrôle de concurrence.

La mémoire transactionnelle proposée se présente comme un ensemble d'objets. Chaque objet de cette mémoire transactionnelle possède deux attributs : un identifiant (String) et une valeur (int).

Le service de mémoire transactionnelle contrôle l'exécution d'un ensemble de transactions définies par une classe abstraite (AbstractTM), et lancées par appel à la méthode newTransaction de cette interface. Chaque transaction effectue une série d'accès, et s'achève par une validation ou un abandon.

Un interpréteur interactif (shell) permet de lancer et d'exécuter pas à pas un ensemble de transactions sur ce service de mémoire transactionnelle. Ces transactions peuvent être contrôlées pas à pas ou spécifiées par un script. Les commandes de cet interpréteur sont les suivantes :

- Opérations transactionnelles
 - new** <transaction> lancer une transaction d'identifiant <transaction>
 - <transaction> **commit** valider une transaction
 - <transaction> **abort** abandonner une transaction
 - <transaction> **read** <t objet> lire un objet
 - <transaction> **write** <t objet> <valeur> affecter une valeur à un objet
- Opérations du shell
 - init** <tm> (t objet,valeur)* choisit un service de mémoire transactionnelle <tm> et en définit l'ensemble des objets transactionnels avec leur valeur ;
 - run** <script> exécute ligne à ligne les commandes contenues dans un fichier script
 - list** liste le contenu de la mémoire transactionnelle
 - help** aide en ligne
 - exit** quitte le simulateur

Par ailleurs, **un mode simulateur** permet d'évaluer l'efficacité des différents services de mémoire transactionnelle sur des scénarios d'exécution qui peuvent lui être fournis. Ce mode simulateur est présenté en section 4.

3 Exemple et scénario d'exécution

3.1 Scénario 1

3.1.1 Politique de contrôle de concurrence pessimiste sans blocage, avec propagation directe : TMPP

a—Quelles seront les transactions validées, bloquées ou abandonnées ?

- La seule transaction validée est : **T3**.
- Les transactions abandonnées sont : **T1** et **T2**.
- Aucune transaction n'est bloquée.

b—Quel sera l'ordre série équivalent obtenu ?

L'ordre série équivalent est :

1. init TMPP (x,0) (y,0) (z,0) (ft1,0) (ft2,0) (ft3,0)
2. (1)– >(2)– >(3)– >(4)– >(6)– >(10)– >(15)– >(16)

c—Quelles seront les valeurs de variables affichées par la commande list ?

- $x = 3$
- $y = 0$
- $z = 0$
- $ft1 = 0$
- $ft2 = 0$
- $ft3 = 1$

d—Vérifiez vos réponses à l'aide du shell.

On vérifie que les résultats sont bien cohérent à l'aide du shell.

3.1.2 politique de contrôle de concurrence par certification avec propagation différée : TMPC

a—Quelles seront les transactions validées, bloquées ou abandonnées ?

- Les transactions validées sont : **T1** et **T2**.
- La seule transaction abandonnée est : **T3**
- Aucune transaction n'est bloquée.

b—Quel sera l'ordre série équivalent obtenu ?

L'ordre série équivalent est :

1. init TMPC (x,0) (y,0) (z,0) (ft1,0) (ft2,0) (ft3,0)
2. (1)– >(2)– >(5)– >(8)– >(11)– >(12)– >(4)– >(7)– >(9)– >(13)– >(14)

c—Quelles seront les valeurs de variables affichées par la commande list ?

- $x = 0$
- $y = 1$
- $z = 2$
- $ft1 = 1$
- $ft2 = 1$
- $ft3 = 0$

d—Vérifiez vos réponses à l'aide du shell.

On vérifie que les résultats sont bien cohérent à l'aide du shell.

3.1.3 réalisant le protocole de verrouillage à 2 phases avec propagation directe : TM2PL

a—Quelles seront les transactions validées, bloquées ou abandonnées ?

- Les transactions validées sont : **T1**, **T2** et **T3**.
- Aucune transaction n'est abandonnées.
- Les transactions bloquées sont : **T1** par **T2** et **T2** par **T3**.

b—Quel sera l'ordre série équivalent obtenu ?

L'ordre série équivalent est :

1. init TM2PL (x,0) (y,0) (z,0) (ft1,0) (ft2,0) (ft3,0)
2. (1) - > (2) - > (3) - > (4) - > (6) - > (10) - > (15) - > (16) - >
(7) - > (9) - > (13) - > (14) - > (5) - > (8) - > (11) - > (12)

c—Quelles seront les valeurs de variables affichées par la commande list ?

- $x = 3$
- $y = 1$
- $z = 1$
- $ft1 = 1$
- $ft2 = 1$
- $ft3 = 1$

d—Vérifiez vos réponses à l'aide du shell.

On vérifie que les résultats sont bien cohérent à l'aide du shell.

3.2 Scénario 2

3.2.1 Politique de contrôle de concurrence pessimiste sans blocage, avec propagation directe : TMPP

- a*—Quelles seront les transactions validées, bloquées ou abandonnées ?
- La seule transaction validée est : **T1**.
 - Les transactions abandonnées sont : **T2** et **T3**.
 - Aucune transaction n'est bloquée.

b—Quel sera l'ordre série équivalent obtenu ?

L'ordre série équivalent est :

- 1.init TMPP (x,0) (y,0) (z,0) (ft1,0) (ft2,0) (ft3,0)
2. (1)– >(2)– >(3)– >(5)– >(8)– >(11)– >(12)

c—Quelles seront les valeurs de variables affichées par la commande list ?

- $x = 0$
- $y = 0$
- $z = 0$
- $ft1 = 1$
- $ft2 = 0$
- $ft3 = 1$

d—Vérifiez vos réponses à l'aide du shell.

On vérifie que les résultats sont bien cohérent à l'aide du shell.

3.2.2 politique de contrôle de concurrence par certification avec propagation différée : TMPC

- a*—Quelles seront les transactions validées, bloquées ou abandonnées ?
- Les transactions validées sont : **T1** et **T2**.
 - La seule transaction abandonnée est : **T3**
 - Aucune transaction n'est bloquée.

b—Quel sera l'ordre série équivalent obtenu ?

L'ordre série équivalent est :

- 1.init TMPC (x,0) (y,0) (z,0) (ft1,0) (ft2,0) (ft3,0)
2. (1)– >(2)– >(3)– >(5)– >(8)– >(11)– >(12)– >(4)– >(7)– >(9)– >(13)– >(14)

c—Quelles seront les valeurs de variables affichées par la commande list ?

- $x = 0$
- $y = 1$
- $z = 2$
- $ft1 = 1$
- $ft2 = 1$
- $ft3 = 0$

d—Vérifiez vos réponses à l'aide du shell.

On vérifie que les résultats sont bien cohérent à l'aide du shell.

3.2.3 réalisant le protocole de verrouillage à 2 phases avec propagation directe : TM2PL

- a*—Quelles seront les transactions validées, bloquées ou abandonnées ?
- Aucune transaction n'est validées.

- Aucune transaction n'est abandonnée.
- Les transactions bloquée sont : **T1**, **T2** et **T3** (Il y'a interblocage)

b— Quel sera l'ordre série équivalent obtenu ?
Interblocage donc pas d'ordre.

c— Quelles seront les valeurs de variables affichées par la commande list ?

- $x = 0$
- $y = 1$
- $z = 0$
- $ft1 = 0$
- $ft2 = 0$
- $ft3 = 0$

d— Vérifiez vos réponses à l'aide du shell.
On vérifie que les résultats sont bien cohérent à l'aide du shell.

4 Évaluation de protocoles

1— Commentaire sur les scénarios fournis :

- **Scénarios 1** : Les 3 transactions sont validées. On remarque une lente durée d'attente.
- **Scénarios 2** : Deux transactions (**T1** et **T2**) sont validées par contre **T3** est abandonnée. Par rapport au temps, on remarque qu'on atteint le temps optimal avec une perte non négligable d'unité de temps à cause de T3 qui est abandonnée.
- **Scénarios 3** : Les 3 transactions sont validées. Par rapport au temps, il n'est pas optimal mais mieux que le premier scénario.

2— Les scénarios permettant de caractériser les situations favorables à chacune des politiques sont :

- **TMPP** : Le scénario le plus favorable pour ce protocole est le cas des transactions non sérialisables.
- **TMPC** : Le scénario le plus favorable pour ce protocole est le cas des transactions sérialisables.
- **TM2PL** : Le scénario le plus favorable pour ce protocole est l'absence ou peu d'accès concurrent au variable car l'utilisation des verrous le ralentit.