

## LOG735 – Systèmes distribués

### *Laboratoire 3 : État global d'un système distribué*

Auteurs : Mohammed Elkanouni, Lévis Thériault, Jean-Philippe Legault (E14)

Réviseurs : Charles-André Bouchard (E10), Jérôme Gagnon (E11)

---

## Résumé

Vous devez appliquer l'algorithme de Chandy-Lamport pour la prise d'état global d'un système bancaire distribué en fonctionnement.

## Durée du laboratoire

Trois séances : 3 juin, 10 juin et 17 juin 2016

## Objectifs

- Familiariser avec la notion d'état global d'un système distribué.
- Implémenter l'algorithme de Chandy-Lamport afin d'obtenir l'état global d'un système distribué.

## Matériel fourni

- Aucun matériel fourni.

## Contexte

Le contexte de l'application servant à explorer les notions du laboratoire consiste en une simulation d'un système transactionnel entre les succursales d'une banque. Au démarrage de la simulation, chacune des succursales possède un montant d'argent spécifié à leur instanciation. Puis, au fil du temps de simulation, les succursales effectueront des transferts d'argent entre elles à un rythme aléatoire. Pour des fins de simplicité, en tout temps, la somme totale d'argent dans le système bancaire est représentée par un entier.

Dans ce contexte, vous devez implémenter une fonctionnalité de capture de l'état global du système bancaire. Ainsi, chaque succursale doit être en mesure de lancer manuellement et aléatoirement une procédure de capture d'état qui affichera, dans l'interface de la succursale, les sommes détenues par chaque autre succursale ainsi que les sommes des transferts en cours de transmission sur les connexions entre succursales.

## Exigences fonctionnelles du système bancaire

Afin de vous aider à diriger votre conception ainsi que vos efforts d'implémentation, voici une série d'exigences fonctionnelles par rapport au système bancaire simulé qui servent aussi à établir des critères pour la correction du laboratoire.

### *Banque*

Vous devez concevoir une entité qui représente la banque supportant le système de transactions simulé. Cette Banque n'offre pas de fonctionnalités concrètes à l'utilisateur de la simulation ; elle doit servir strictement à centraliser les connexions des Succursales afin de les relier entre elles.

BANQUE-01 : La Banque doit accepter la connexion réseau d'une Succursale et l'intégrer à sa liste de Succursales.

BANQUE-02 : La Banque doit envoyer à une Succursale sa liste de Succursales avec leurs coordonnées réseau afin qu'une Succursale puisse former une connexion persistante avec chacune des autres Succursales de la Banque.

BANQUE-03 : La Banque doit garder en mémoire la somme totale d'argent initiale contenue dans l'ensemble des Succursales. Ainsi, si trois Succursales s'y connectent avec des montants de 2000\$ chacun, la Banque doit conserver l'information que 6000\$ ont été insérés dans le système bancaire.

BANQUE-04 : La Banque doit permettre l'insertion en cours de simulation de nouvelles Succursales sans altérer le bon fonctionnement du système. **NOTE : Cependant, vous n'avez pas besoin de gérer le retrait d'une Succursale.**

BANQUE-05 : Après chaque ajout de succursale, la Banque doit afficher le nouveau montant ajouté par la succursale et le nouveau montant total présent dans l'ensemble des succursales.

BANQUE-06 : La Banque attribut à chaque succursale, lorsqu'ils sont ajoutées dans le système, un numéro unique (1, 2, 3, 4,...) pour les identifier. Ce numéro est renvoyé à la succursale et sert à l'identifier tout au long de la simulation. La banque est responsable de s'assurer que ce numéro est unique.

### *Succursale*

Vous devez concevoir une entité qui représente une instance de succursale associée à une Banque. L'utilisateur de la simulation interagira strictement avec les interfaces de ces Succursales.

SUCCURSALE-01 : La Succursale doit pouvoir établir une connexion réseau à une Banque, à partir du poste de la Banque ou d'un autre poste.

SUCCURSALE-02 : La Succursale doit pouvoir démarrer sa présence dans la simulation avec un montant d'argent passé en paramètre (dans la console) par l'utilisateur de la simulation. Ce montant d'argent doit être positif et entier ; il n'est pas nécessaire de gérer les fractions de dollars.

SUCCURSALE-03 : La Succursale doit établir une connexion persistante avec chacune des autres Succursales du système bancaire et utiliser cette connexion pour recevoir et envoyer des transferts d'argent.

SUCCURSALE-04 : La Succursale doit afficher la liste des autres Succursales connectées. Lorsqu'une nouvelle Succursale est ajoutée, on affiche à nouveau la liste de tous les Succursales

SUCCURSALE-05 : La Succursale doit effectuer périodiquement un transfert d'argent en direction d'une autre Succursale sans intervention de l'utilisateur de la simulation. Une pause variable entre 5 et 10 secondes doit être imposée entre chacun des transferts. Dans chacun des transferts, la Succursale doit envoyer une fraction aléatoire et arrondie à l'entier le plus près de la somme d'argent qu'elle possède à cet instant ; elle ne doit pas envoyer une somme d'argent qu'elle ne possède pas. Si la Succursale ne possède plus d'argent, elle ne doit pas effectuer de transfert.

SUCCURSALE-06 : La Succursale doit permettre l'envoi manuel d'une transaction par la console à une autre Succursale. Les paramètres de la transaction doivent être validés. Encore une fois, une Succursale ne doit pas envoyer plus d'argent qu'elle n'en possède.

SUCCURSALE-07 : La Succursale doit permettre l'introduction manuelle d'une erreur pour simuler une perte d'argent. L'erreur peut provenir d'un mauvais Transfert et/ou simplement d'un problème avec le solde de la succursale.

### ***Transfert d'argent***

Dans le contexte de la simulation, un Transfert d'argent consiste en un message qui contient au minimum la somme d'argent transférée d'une Succursale à une autre. ATTENTION : l'implémentation de l'algorithme de Chandy-Lamport **nécessitera** l'ajout d'informations supplémentaires dans les messages transmis.

TRANSFERT-01 : Le Transfert d'argent doit contenir la somme d'argent transférée entre deux Succursales sous la forme d'un nombre entier positif.

TRANSFERT-02 : Aucun Transfert d'argent ne doit permettre à une Succursale d'obtenir un solde négatif.

TRANSFERT-03 : Pour reproduire de la latence sur le réseau (et tester si l'algorithme de Chandy-Lamport est bien implémenté), vous devez simuler un délai de 5 secondes lors de chaque Transfert d'argent. C'est-à-dire, lors d'une transaction, la Succursale affiche le Transfert à l'écran, retire instantanément le montant de son solde mais attend 5 secondes avant d'envoyer le Transfert à l'autre succursale.

## Exigences fonctionnelles pour la capture de l'état global

La capture de l'état global est un procédé qui doit permettre à l'utilisateur d'obtenir les informations sur l'ensemble du système à partir de n'importe quelle Succursale. Cet énoncé ne vous expliquera pas comment implémenter l'algorithme de Chandy-Lamport devant supporter le procédé : vous devez vous fier au contenu du cours afin de bien comprendre l'algorithme.

ETAT-01 : Une Succursale doit permettre à l'utilisateur de la simulation de lancer une capture de l'état global du système bancaire à partir de celle-ci. Cela peut prendre la forme, par exemple, d'un bouton graphique ou d'une instruction lancée en ligne de commande.

ETAT-02 : Toutes les Succursales doivent périodiquement lancer une capture de l'état global sans intervention de l'utilisateur de la simulation. Une pause variable entre 10 et 30 secondes doit être imposée entre chacun des lancements. Ce mode automatique doit coexister avec le mode manuel de l'exigence ETAT-01 sans altérer le bon fonctionnement du système.

ETAT-03 : Le système bancaire doit supporter plusieurs captures de l'état global simultanées et asynchrones. En d'autres termes, le fait qu'une ou plusieurs captures soient présentement en cours à travers le système ne doit pas influencer le fonctionnement d'une nouvelle capture.

ETAT-04 : Une Succursale doit être en mesure de fonctionner sans interruption ni changements pendant qu'une capture de l'état global est en cours à partir de celle-ci. Le processus doit être transparent et ne doit pas affecter significativement la performance du système.

ETAT-05 : Le résultat de la capture doit afficher à l'utilisateur de la simulation les informations selon le format suivant. Par exemple, dans le cas où trois succursales sont connectées entre elles et que la capturée est lancée à partir de la #1 :

*Succursale d'origine de la capture : #1*

*Succursale #1 : 1400\$*

*Succursale #2 : 1800\$*

*Succursale #3 : 2200\$*

*Canal S1-S2 : 400\$*

*Canal S1-S3 : 200\$*

*Somme connue par la Banque : 6000\$*

*Somme détectée par la capture : 6000\$*

*ÉTAT GLOBAL COHÉRENT*

**[NOTE : Assurez-vous de n'afficher qu'une fois l'information de chaque canal. Il ne faut pas voir « S1-S2 » et « S2-S1 »...]**

Dans le cas où une incohérence est détectée, la dernière partie de l'affichage doit plutôt être produite comme suit :

*Somme connue par la Banque : 6000\$*

*Somme détectée par la capture : 5800\$*

*ÉTAT GLOBAL INCOHÉRENT*

## Étapes d'implémentation et questions

La pondération des étapes est indiquée entre crochets. Les instructions sont écrites en format régulier. Les questions sont écrites en format souligné. **Ne copiez pas les instructions dans le rapport ; copiez les questions auxquelles vous répondez.**

1. [45%] Concevez et implémentez le système bancaire distribué en fonction des exigences fonctionnelles spécifiées pour le système bancaire **seulement**.  
Produisez un diagramme de classes représentant l'ensemble du système avec, en annexe, toutes informations complémentaires requises pour une bonne compréhension de votre conception. Produisez aussi un diagramme de séquence illustrant le comportement du système entre le moment auquel une Succursale démarre un processus de transfert d'argent et le moment où la Succursale de destination ajoute la somme d'argent reçue à sa propre somme. Votre conception illustrée à cette étape ne doit pas tenir compte des exigences reliées à la capture de l'état global : considérez cette étape comme une première itération du système.
2. [45%] Concevez et implémentez la capture de l'état global en fonction des exigences fonctionnelles spécifiées.  
Produisez un nouveau diagramme de classes représentant l'ensemble du système qui, cette fois-ci, illustre adéquatement les changements apportés à l'itération précédente. Ajoutez en annexe toutes informations complémentaires requises pour une bonne compréhension de votre conception. Produisez aussi un diagramme de séquence illustrant le comportement du système entre le moment auquel l'utilisateur de la Succursale démarre une capture de l'état global et le moment où le résultat de la capture est affiché.
3. [10%] Discutez des améliorations requises à votre solution dans le cas où nous voudrions assurer une tolérance aux fautes suivantes :
  - a. Omission de réception d'un transfert d'argent : S1 envoie un transfert à S2, mais un bris de réseau survient et S2 ne reçoit jamais le transfert. S1 ne doit pas perdre le montant qui n'a pas été transféré.
  - b. Déconnexion d'une Succursale du système bancaire : S3 tombe en panne et ne répond plus aux autres Succursales. La Banque lance une capture d'un état global sur les Succursales encore actives et ajuste sa somme totale connue en conséquence.

## Contraintes d'implémentation

- Le laboratoire doit être implémenté à l'aide du langage de programmation Java.

## Correction interactive

Le lendemain de la remise, une évaluation sommaire et **obligatoire** de la fonctionnalité de l'application sera faite pour chaque équipe. Les deux étudiants de l'équipe doivent être présents ; en cas d'absence non-justifiée, les points alloués au bon fonctionnement de l'implémentation seront retirés.

## Pénalités de correction

Les pénalités suivantes ainsi que leur valeur sont applicables comme suit :

- Fautes de français et erreurs de rédaction : -0.5% par erreur, maximum 20 fautes (-10%);
- Documentation du code source : -5% si insuffisante, -10% si pratiquement inexistante;
- Erreurs de compilation : perte de **tous** les points alloués au bon fonctionnement;
- **Retard de remise non-justifié : note de zéro automatique et non négociable;**
- **Non-respect des normes de remise : note de zéro automatique et non négociable.**

## Normes de rédaction et de remise

- Les normes de rédaction contenues dans la présentation « Normes de rédaction et de remise » s'appliquent dans leur intégralité.
- La remise électronique doit contenir l'application améliorée avec tous les *packages* nécessaires.

## Échéances

- Lors de la troisième séance (**17 juin 2015**), vous devrez présenter au chargé de laboratoire une version intermédiaire de votre solution qui est en mesure d'effectuer des Transferts entre Succursales ainsi que les autres caractéristiques associées à cette tâche (par exemple : l'ajout d'une Succursale à la Banque, l'établissement des connexions entre chaque Succursales, etc.). La présence de chaque membre de l'équipe est obligatoire.
- Remise électronique (code source + rapport électronique) : **Le 22 juin avant 23h59 (ATTENTION C'EST UN MERCREDI).**
- Remise papier du rapport lors de la correction interactive.
- Correction interactive : **Le 23 juin à 13h30 (ATTENTION C'EST UN JEUDI).**