

Haute école d'ingénierie et de gestion du Canton de Vaud
Département TIC
Laboratoire de programmation répartie PRR

Temps à disposition : 10 périodes (travail débutant le jeudi 29 octobre 2015, semaine 6)

Objectifs

- Comprendre le fonctionnement d'un algorithme d'exclusion mutuelle réparti;
- Etendre l'algorithme de Ricart et Agrawala pour maintenir la cohérence d'une variable partagée sur un ensemble de sites.

Énoncé

Partager une donnée parmi un ensemble de sites est un problème qui peut se résoudre par le biais d'un algorithme d'exclusion mutuelle. Dans ce laboratoire, nous souhaitons modifier l'utilisation prévue de l'algorithme de *Ricart et Agrawala*.

Chaque site détient une variable entière qui doit être cohérente. Les processus usagers peuvent faire 2 opérations sur cette variable : consulter sa valeur et modifier sa valeur.

La consultation revient à obtenir la valeur locale la plus récente. Par contre, une modification se réalise en 3 étapes :

1. Obtenir l'exclusion mutuelle sur la variable;
2. Modifier la valeur en section critique;
3. Informer tous les autres sites de la nouvelle valeur quand la section critique est libérée.

Pour l'algorithme de *Ricart et Agrawala*, la dernière étape se fait avec un message supplémentaire pour tous les sites qui n'ont pas été différés par celui qui sort de sa section critique.

Travail à faire

Chaque site de l'environnement exécute 2 machines virtuelles. La première est un gestionnaire qui maintient la cohérence de la variable partagée sur le site. C'est cette VM qui implémente l'algorithme de *Ricart et Agrawala* et qui détient une copie de la variable partagée. La seconde VM exécute les tâches applicatives. Ces tâches fournissent un menu textuel simple permettant d'effectuer les 2 opérations suivantes :

1. afficher la valeur de la variable partagée obtenue par une requête au gestionnaire du site (opération de consultation);
2. obtenir la permission de modifier la variable partagée, puis réalisation de cette modification par une saisie au clavier.

La première opération n'est jamais bloquante et pourra être effectuée par toutes les tâches applicatives qui ne sont ni en attente d'une exclusion mutuelle et ni en cours de modifier la variable partagée. C'est uniquement la seconde opération qui nécessite l'obtention de l'exclusion mutuelle, et afin de vérifier le fonctionnement de votre implémentation, vous devez afficher la valeur de la variable partagée en section critique avant sa modification. Aussi, après libération de la section critique, la nouvelle valeur de la variable sera transmise aux autres sites formant l'environnement.

Rappelons que l'algorithme de Ricart et Agrawala procède par obtention d'autorisations, et qu'un site ne peut accéder en section critique qu'après avoir obtenu toutes les autorisations des autres sites. Quand un site est en section critique, il doit différer ses autorisations. Ainsi, la valeur modifiée en section critique peut être jointe aux messages d'autorisation différés et émis par le site sortant de sa section critique de manière à informer les sites en attente de sa nouvelle valeur. Pour tous les autres sites, ceux n'attendant pas d'autorisation, un message particulier peut leur être adressé. Remarquons qu'un message contenant la mise à jour de la variable partagée doit comporter une estampille et qu'un site ne peut pas écraser la valeur actuelle de sa variable partagée par une valeur plus ancienne.

Hypothèses

1. Le nombre de sites partageant la donnée n'est pas connu, mais peut être introduit comme paramètre sur la ligne de commande. Une fois initialisé, le nombre de sites demeure constant.
2. Ces sites et le réseau qui les interconnecte sont entièrement fiables.
3. La communication entre les différentes VM se réalisera uniquement par UDP.
4. Il peut y avoir plusieurs tâches applicatives sur un site.
5. Il n'y a qu'une seule variable globale à partager sur l'ensemble des sites.
6. Tous les messages échangés devront être le plus petit que possible.
7. Afin de simplifier l'initialisation des gestionnaires d'exclusion mutuelle, vous êtes libre de prendre toutes les hypothèses qui simplifient le problème. Typiquement, vous devez lancer toutes les VM maintenant la cohérence de la variable globale avant de lancer les tâches applicatives.

Remarques sur le travail à effectuer

- Vous devez nous rendre un listage (papier) complet de vos sources (fichiers sources Java).
- La description de l'implémentation, ses différentes étapes et toute autre information pertinente doivent figurer dans les programmes rendus. Aucun rapport n'est demandé.
- Inspirez-vous du barème de correction pour connaître là où il faut mettre votre effort.
- Vous pouvez travailler en équipe de deux personnes au plus.
- Une démo de votre programme pourra être demandée.

Barème de correction

Conception (structure et décomposition)	15%
Portabilité sur un autre environnement réparti, réutilisation du code dans un projet plus conséquent, robustesse	15%
Exécution et fonctionnement	20%
Codage (choix des variables, opérations, lisibilité, localité de référence, etc.)	15%
Documentation des en-têtes (programme et méthodes)	25%
Commentaires au niveau du code (qualité et complétude)	10%