UPMC
SORBONNE UNIVERSITÉS

TME ARA 2017–2018
Points de reprise

Jonathan Lejeune

Objectifs

L'objectif de ce TME est d'analyser un code qui simule un algorithme de points de reprise sur lequel s'exécute une application distribuée. En suivant le modèle qui vous a été donné il vous sera ensuite demandé d'implémenter un autre algorithme dans PeerSim.

Installation

Créez dans votre projet Eclipse un package *tme3* et copiez-y dans le dossier correspondant les fichiers fournis dans vos ressources TME. Rafraîchissez et assurez-vous que la compilation se passe bien (absence de croix rouge).

Exercice 1 – Analyse de l'application

Dans cet exercice nous nous concentrerons uniquement sur l'application concernée sans encore tenir compte des mécanismes de sauvegarde et de restauration des points de reprises.

Question 1
Décrire ce que fait l'application et l'algorithme utilisé. Quels sont les avantages de cet algorithme ?

Question 2
Expliquez le rôle des méthodes de l'interface Checkpointable.

Question 3
Comment sont déclenchés les appels à Request et Release ? Pourquoi ces événements sont-ils associés à un identifiant logique ?

Question 4
Quel est l'impact sur l'application du rapport temps entre deux requêtes/temps de section critique ?

Question 5

Écrivez un fichier de configuration permettant d'exécuter l'application. Vous vous baserez sur

— un protocole de transport classique avec une latence variant entre 5 et 100 unités de temps

— 5 nœuds

— la considération de plusieurs valeur du rapport temps entre deux requêtes/temps de section critique : 1, 5, 10 et 0.1

Il vous est possible de décommenter certains affichage afin d'avoir une trace d'exécution plus explicite de l'algorithme applicatif.

Question 6

Ajoutez au fichier de configuration le module de contrôle CrashControler en spécifiant que le noeud 4 est défaillant avec une probabilité de 1 à la date 10000. Pour l'instant ne pas le lier au checkpointer.

Question 7

Testez la simulation. Dans quel cas, la panne d'un nœud n'aurait aucun impact sur l'application ?

Exercice 2 – Mécanisme de point de reprise

Pour palier le problème précédent nous allons intégrer un protocole de recouvrement de point de reprise déclenché par le contrôleur CrashControler au moment où un nœud tombera en panne. Les algorithmes de sauvegardes et de recouvrement de points de reprise sont déjà codés dans le fichier Checkpointer.java.

Question 1

Décrivez le mécanisme permettant de faire un point de sauvegarde. Quel intérêt de ne pas avoir une période de sauvegarde strictement fixe pour tous les nœuds ?

Question 2

Décrivez l'algorithme de sauvegarde et en déduire si c'est un algorithme coordonnée, non coordonné ou de journalisation.

Question 3

Indépendament de l'application, quelles sont les données sauvegardées à chaque point de reprise ?

Question 4

Quelle(s) structure(s) de données permettent de simuler la sauvegarde des points de reprise sur un support stable. Pourquoi avoir choisi ce(s) type(s) de structure ?

Question 5

Pourquoi la classe Checkpointer implante-t-elle un protocole de transport ?

Question 6

Sachant que le Checkpointer est supposé se basé lui-même sur un transport à cannaux FIFO (voir FIFOTransport.java), que se passerait-il si les messages Wrapping n'étaient pas utilisés ?

Question 7

Décrivez l'algorithme de recouvrement ainsi que le rôles des différents types de message. Quelles sont les principales étapes de cet algorithme ? Quel est l'impact sur l'application ?

Question 8

Expliquez le code de la méthode traitant la réception d'un message AskMissingMess. Expliquez pourquoi l'hypothèse d'avoir des canaux FIFO est nécessaire.

Question 9

Écrivez un fichier de configuration incluant la couche protocolaire du Checkpointer. Attention le Checkpointer considère que les cannaux sont FIFO : n'oubliez pas d'utiliser le protocole de transport FIFO qui vous a été fourni.

Question 10

Vérifiez qu'au moment de la panne d'un nœud, l'algorithme de recouvrement permet au système de revenir à un état cohérent et de rejouer son exécution. Il vous sera possible d'évaluer l'impact de la fréquence de sauvegarde des points de reprises.

Question 11

En vous inspirant du code qui vous a été fourni, codez un nouveau Checkpointer qui implémente l'algorithme de Chandy-Lamport.