

Projet de système distribués

Stéphane Pouget

30 avril 2018

1 Topologie

J'ai créé un anneau car c'est une topologie simple et efficace. On peut supprimer et ajouter un nœud sans aucun problème. L'anneau n'est pas efficace pour la transmission de message à une personne précise mais ici ce n'est pas le but recherché de ce fait cette topologie convient parfaitement.

Chaque nœud connaît son fils et la taille de l'anneau. Avec cela je peux tout faire. Si je veux ajouter un nœud après un autre nœud je lui dis que son nouveau fils c'est ce nouveau nœud et son ancien fils devient le fils du nouveau nœud. C'est l'inverse lorsqu'on supprime un nœud.

2 Structure du code

J'ai un seul fichier qui gère tout.

ring_node : Fonction principale qui s'occupe de la gestion de la ring. Cette fonction tourne non stop et attend de recevoir les messages pour qu'elle puisse agir. Ainsi elle s'occupe de la création de l'anneau, l'ajout d'un nœud, la suppression d'un nœud, le broadcast, et l'envoi des données.

data : C'est la fonction qui s'occupe du dictionnaire qui contient les informations échangées. Ce dictionnaire a pour clé l'UUID et a pour valeur le message.

build : Cette fonction permet de lancer la création d'un anneau de dimension $N+1$.

loop : Fonction auxiliaire pour la création de l'anneau, elle vérifie juste que l'anneau est bien créé.

broad : Lance le broadcast

add : Permet d'ajouter un nœud. Je refais un broadcast pour régler la taille de la ring est en même temps vérifier que cela a bien marché.

kill : Permet de supprimer un nœud.

send : Permet d'envoyer un message avec un UUID.
erase : Permet de supprimer un message.
getvalue : Permet d'avoir un message connaissant l'UUID.

3 Usage

Compiler : `c(ring)`.
Création anneau : `ring :build(N)`. avec N nombre de nœud.
Broadcast : `ring :broad(PID)`.
Ajout d'un nœud : `ring :add(PID)`.
Suppression d'un nœud : `ring :kill(PID)`.
Avoir un message : `ring :getvalue(Key, PID)`.
Supprimer un message : `ring :erase(Key, PID)`.
Envoyer un message : `ring :send(Message, UUID, PID)`.