Simulation d'algorithmes d'équilibrage de charge dans un environnement distribué

Identifications des besoins

Kevin Barreau Guillaume Marques Corentin Salingue

28 janvier 2015

Résumé

Ce document dégage une première identification des besoins. Il s'agit d'un document support pour l'élaboration du cahier des charges

1 Définition du projet

1.1 Définitions

L'expansion, au cours des deux dernières décennies, des réseaux et notamment d'Internet a engendré une importante création de données, massives par leur nombre et leur taille. Stocker cette information sur un seul point de stockage (ordinateur par exemple) n'est bien sûr plus envisageable, que ce soit pour des raisons techniques ou pour des raisons de sécurité (pannes potentielles par exemple). Pour cela des systèmes de stockages dit distribués sont utilisés en pratique afin des les répartir sur dfférentes unités de stockages.

Définition Un environnement distribué est constitué de plusieurs machines (ordinateurs), appelées *noeuds*, sur lesquelles sont stockées des données.

Définition Une donnée est une suite binaire de 0 et de 1 dont le contenu n'est pas important pour l'application.

Le client souhaite répartir toutes ces données de manières équitable entre les noeuds. De plus, il souhaite que ces données soient accessible afin de pouvoir les requêter et récuperer de l'information.

Définition Une requête est un message envoyé à une machine (ou plusieurs machines) afin de récuperer de l'information sur des données. Nous noterons que la nature de l'information est inutile pour le bon fonctionnement de l'application.

Pour répartir toutes ces données, notre client a développer de nouveaux algorithmes d'équilibrage de charges et de réplication qu'il souhaite tester dans un environnement distribué.

1.2 Finalité

Nous devons développer une solution logicielle permettant de tester ces nouveaux algorithmes d'équilibrage de charge et de réplication.

2 Besoins fonctionnels

L'environnement de simulation voulu est un système distribué constitué de n noeuds de stockage dans lequel on souhaite stocker m objets.

2.1 Environnement distribué

Un environnement distribué est constitué de plusieurs machines, appelées *noeuds*, sur lesquelles sont stockées les données et sont distribuées les tâches à effectuer, appelées *requêtes*.

Utilisation de Cassandra. Pourquoi ?

— Le client est plus à l'aise avec cette solution (il a quelques connaissances sur le logiciel)
— Pourquoi pas une autre BDD ? (développer)

2.2 Gestion des noeuds

— Creer un noeud
— Supprimer un noeud (inutile d'après Kévin)s
— Sauvegarder un ensemble de noeuds avec leurs objets
— Importer un ensemble de noeuds avec leurs objets afin de simuler sur un environemment précedemment crée.

Question Quelle format pour le stockage d'un ensemble de noeuds ?
Question Combien de noeuds maximum ?
Question Position des noeuds sur le ring ?

2.3 Gestion des objets

Question Sur quel type d'objet allons-nous travailler ?

— Creer un objet
— Supprimer un objet
— Gérer la popularité d'un objet
  — Implémenter l'algorithme d'approximation Space-Saving Algorithm
  — Ajouter un vecteur de taille n à chaque noeud dans le lequel on stockera les objets les plus populaires
  — Protocole de communication pour le calcul des objets les plus populaires **(Modifier gossip ? Nouveau ?)**

Question Gestion de la popularité : quelle période (fixe ou fenêtre glissante) ?
Question Position de l'objet sur le noeud ?
Note Bien définir l'algorithme.

2.4 Gestion des requêtes

— Creer une requête
— Envoyer une requête sur le réseau
— Sauvegarder un jeu de requête
— Importer un jeu de requête

Question Quelle format pour le stockage d'un jeu de requêtes ?

2.5 Visualisation des données

— Temps de réponse moyen sur les requêtes passées.
— Charge d'un noeud
— Popularité des objets

Note Bien définir ces items.