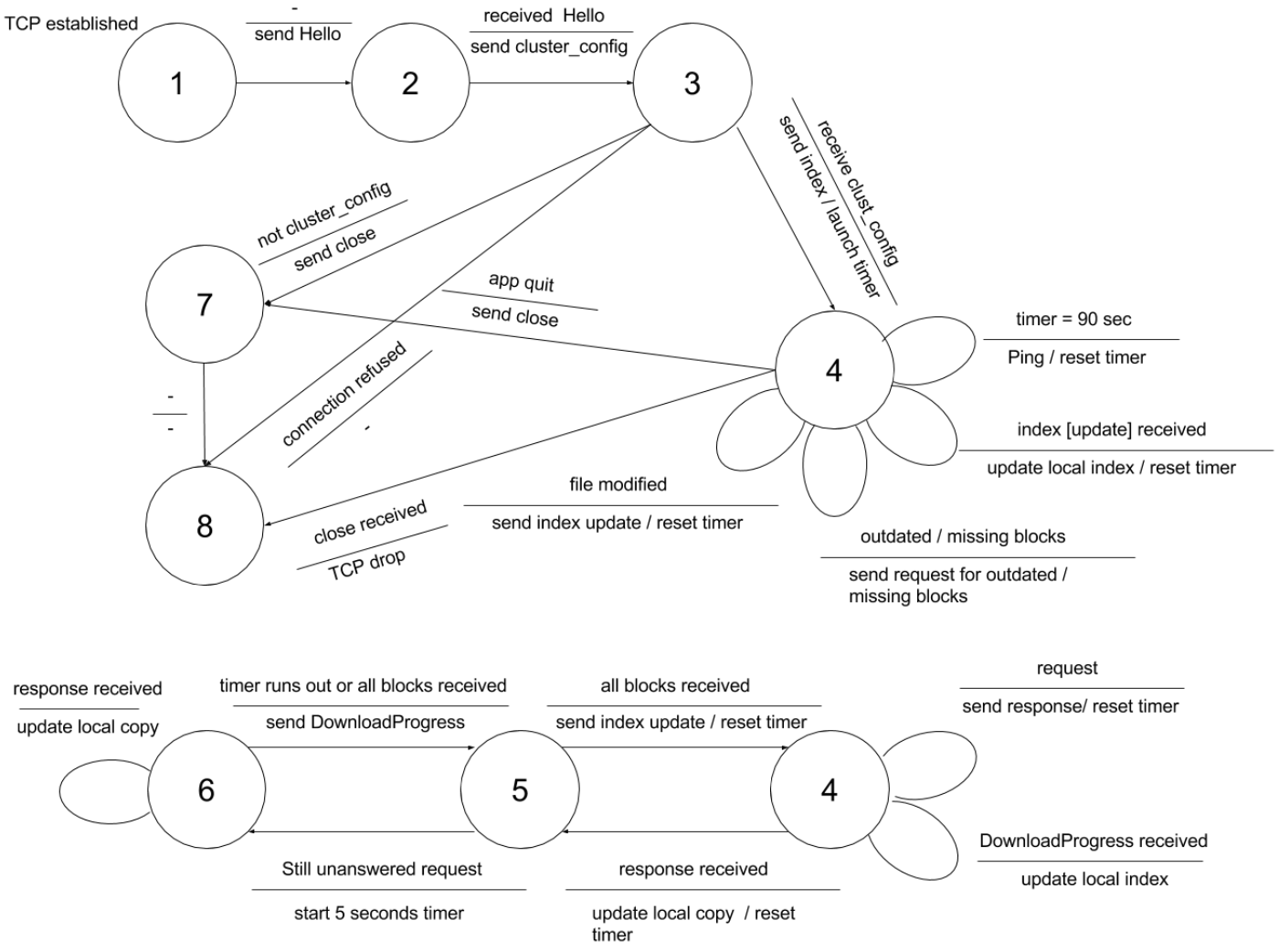
Modélisation et analyse du protocole BEP 1.0

Auteurs : Tournier Vincent, Da Silva Marques Gabriel

Date : 1 Décembre 2017

Graphe d'états

Lors de notre étude du Block Exchange Protocol nous sommes arrivés à obtenir un graphe qui devrait permettre l'exécution du protocole en répondant correctement à toutes les situations.



Le petit groupe de trois états qui se sont perdus en bas est en fait une extension de l'état 4. Tout ne rentrait pas dans le graphe principal donc on a du improviser. Par ailleurs, Syncthing étant destiné à

être distribué sur chaque machine, ce graphe d'état fonctionne que le soit dans un modèle distribué ou dans une architecture "client/serveur", bien que ce modèle ne doive pas être appliqué à BEP.

Diagrammes de temps

Suite à ça nous avons dessiné les trois diagrammes ci-dessous, permettant de visualiser une séquence d'échanges type entre le client et le cluster (Figure 1) et deux échanges comportants chacun une erreur de transmission. (Figure 2 et Figure 3).

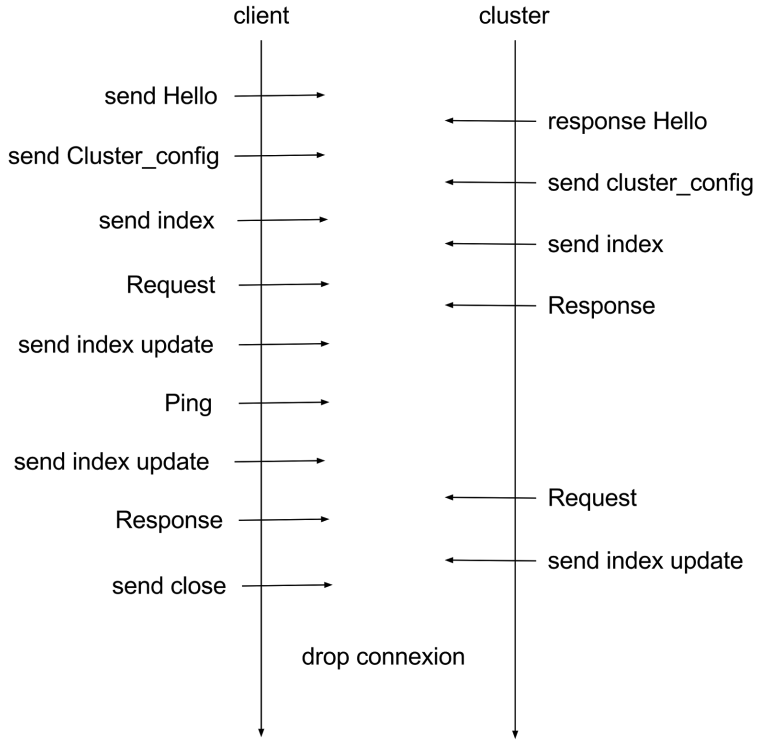


Figure 1 Connexion complète sans erreur

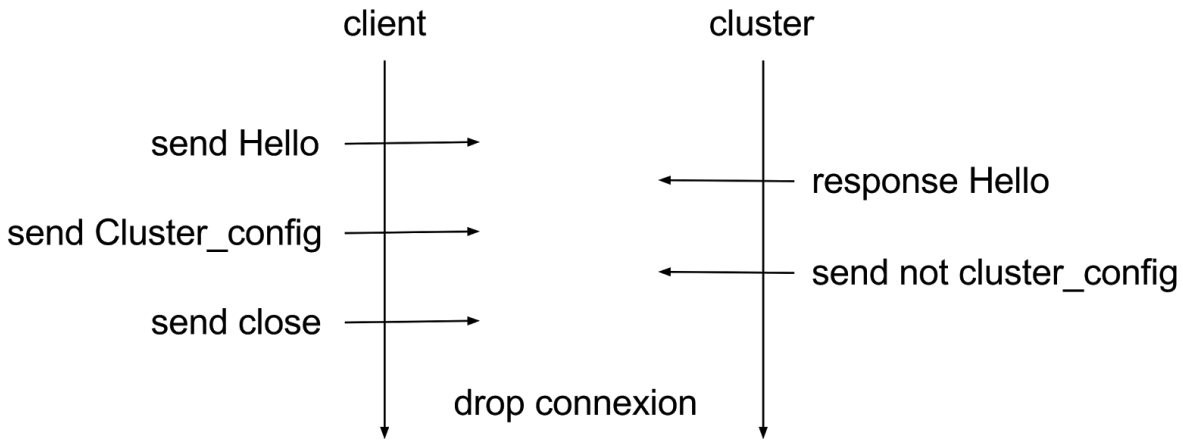


Figure 2 Le premier message reçu après Hello n'est pas un cluster_config

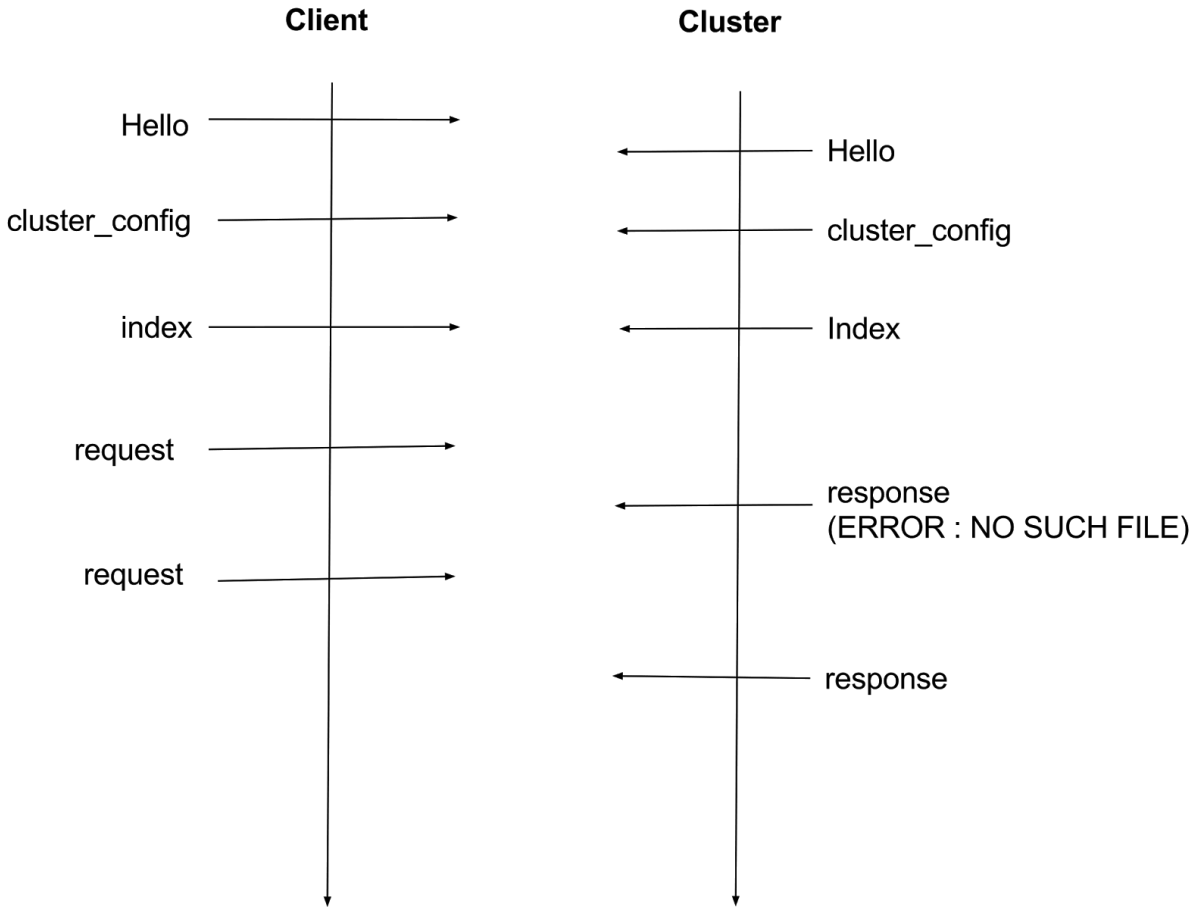
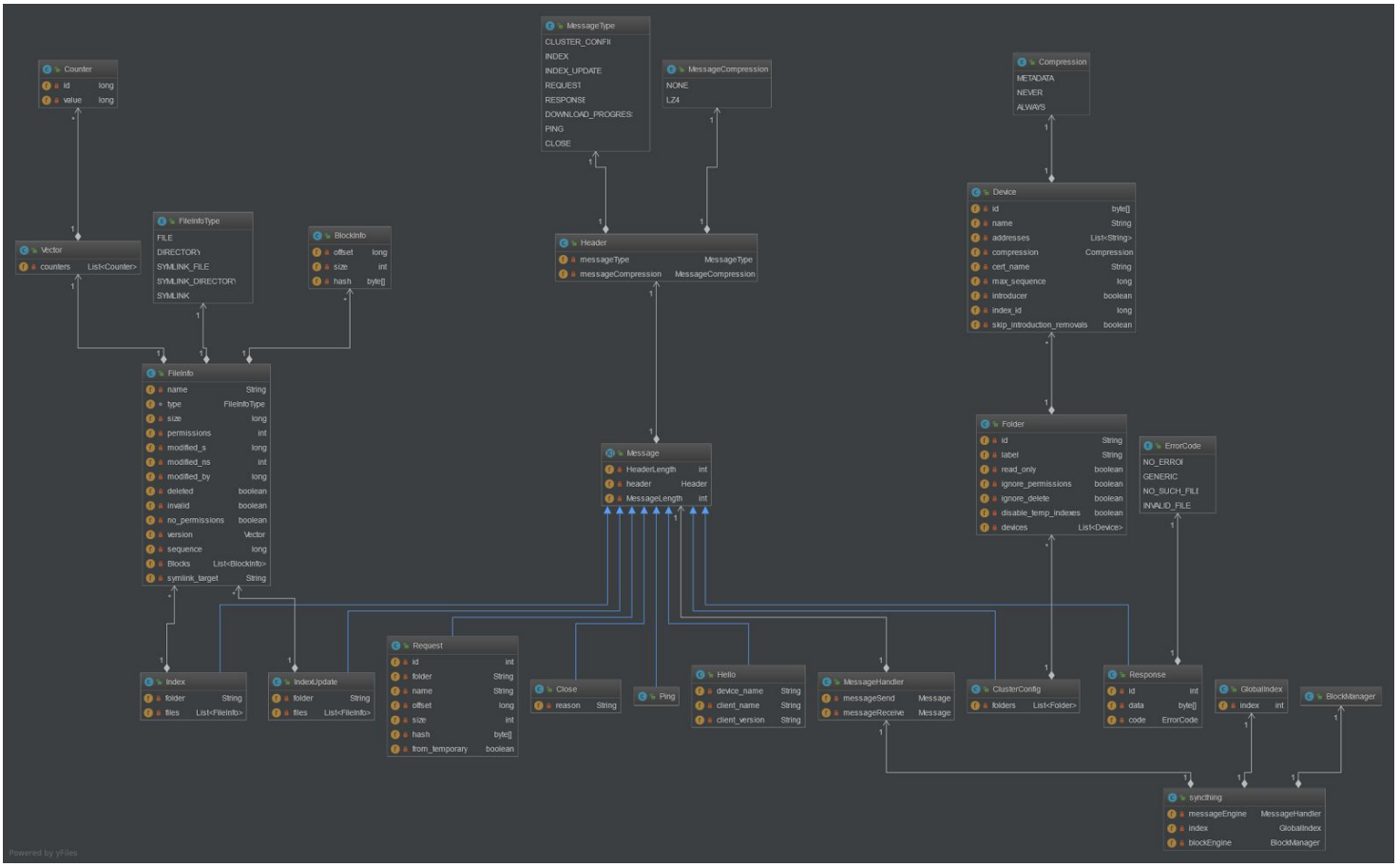


Figure 3 Une request d'un fichier qui n'existe pas

Diagramme de classes

Pour le diagramme de classes nous avons utilisé intellij pour représenter les classes. (C'est sûr, Java c'est pas Python 3... mais c'était juste le temps de faire le diagramme ! On implémentera bien en python)



Le texte est relativement illisible, c'est pourquoi cette image est disponible en brut dans notre repo git.

Les classes des messages seront générées via protobuf et le coeur du programme sera la classe syncthing qui disposera d'un gestionnaire de message, un gestionnaire pour le système de fichier permettant de surveiller les modifications de la copie locale et de les mettre à jour, et d'un gestionnaire d'index qui contiendra l'index global glané au fil des échanges.

Par ailleurs, cette présentation du protocole BEP ne tient pas compte des DownloadProgress. En effet, ceux-ci ne devant pas être implémentés, et sans réponse de votre part, nous avons cru juste de ne pas les inclure dans notre étude.