

UFR Sciences et Techniques, Université de Nantes
Projet de fin d'études

Utilisation concrète de Logoot

Adrien Bougouin
Ronan-Alexandre Cherrueau
Adrien Drouet
Alban Ménager
Alexandre Prenza

Année universitaire 2011-2012
Master ALMA

Table des matières

Introduction	3
1 Gestion de projet	4
2 Analyse	5
2.1 Peer-to-Peer	5
2.1.1 Gossip	6
2.1.2 Jxta	6
3 Conception	7
4 Réalisation	8
Conclusion	9
Bibliographie	10

Introduction

De nos jours, le travail collaboratif possède une place importante. En effet, le monde de l'entreprise et de la recherche oblige de plus en plus le travail à distance entre plusieurs personnes en même temps. Une aide pour ce travail est bien entendu les vidéos-conférences dans lesquelles plusieurs personnes distantes peuvent communiquer en direct et sans problème. Pouvoir éditer des documents en même temps et par tous est un atout majeur dans le travail collaboratif. C'est ce que nous propose *Google Docs*.

En effet, *Google Docs* permet à des utilisateurs de pouvoir travailler sur un même document, en même temps et en voyant les modifications effectuées par les autres utilisateurs directement sur votre écran. Cependant, cette solution possède quelques limites, notamment du fait que le document partagé est centralisé sur un serveur distant et est donc retransmis à tous les autres utilisateurs.

La solution apportée à cette lacune est de distribuer le document partagé en utilisant un réseau peer-to-peer¹, de ce fait, chaque utilisateur possède l'ensemble du document et seulement les modifications apportées à ce dernier seront transmis aux autres utilisateurs qui se mettront à jour. Ceci permet de limiter les risques de perte du document (chacun possède le document et non pas un serveur centralisé) et les modifications apportées à ce dernier seront transmises beaucoup plus rapidement.

L'approche de *Logoot* ne limite pas le nombre d'utilisateur ni le nombre d'édition du document partagé tout en garantissant la causalité, la consistance et la préservation de l'intention.

Ce document présente *Logoot*, sa description, son utilisation ainsi que l'outil développé afin de faire du travail collaboratif en utilisant l'approche *Logoot*.

1. Réseau pair-à-pair

Chapitre 1

Gestion de projet

Chapitre 2

Analyse

2.1 Peer-to-Peer

Un réseau peer-to-peer est un modèle où chaque client est aussi serveur. En effet, dans un réseau peer-to-peer, l'information est diffusée au travers du réseau grâce aux clients qui relaient l'information. Comme représenté en figure 2.1 page 5, le réseau est décentralisé, chaque ordinateur joue un double rôle de client et de serveur. L'avantage d'un tel réseau est qu'il soit décentralisé, en effet, la vie d'une ressource ne repose pas sur un seul serveur (comme c'est le cas dans un réseau classique) mais bien sûr tous les éléments du réseau qui la possède. Ainsi, même si l'un des noeuds cesse d'émettre, la ressource reste disponible via les autres noeuds qui la possède.

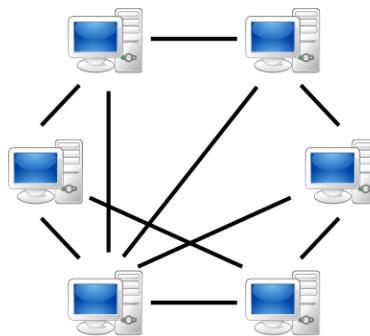


FIGURE 2.1 – Exemple de réseau peer to peer

2.1.1 Gossip

Gossip est un protocole de communication basé sur l'idée d'*épidémie* ou de rumeur : une personne lance une rumeur à ses voisins qui eux-même la disent à leurs voisins, etc au bout d'un certain temps, toutes personnes connaissent la rumeur lancée par une celle d'origine.

L'avantage d'utiliser *Gossip* est qu'il permet de fonctionner dans un réseau non structuré. En effet, en utilisant *Gossip*, il n'est pas nécessaire de connaître la structure du réseau pour pouvoir fonctionner.

On suppose que l'on souhaite faire la recherche d'un ressource X qui se trouve sur l'un des postes du réseau, le principe du protocole est le suivant :

1. Chaque noeud possède une liste de *peer* dans le réseau.
2. À intervalle régulier, le poste prend un peer au hasard dans la liste et lui envoie le message.
3. Le poste qui reçoit le message vérifie s'il possède la ressource : si c'est le cas, il répond, sinon il recommence depuis l'étape 2 et ainsi de suite.

De cette façon, en un temps logarithmique, chaque noeud du réseau aura reçu le message et l'initiateur de la recherche aura donc la réponse. À chaque "tour", le nombre de personne renvoyant le message est doublé ce qui rend le protocole très robuste car, même si un message est perdu en cours de route, le noeud qui aura manqué ce message, le recevra de la part d'un autre peer.

Ce protocole est la solution pour le projet. En effet, le réseau sur lequel les personnes font le travail collaboratif n'est pas connu et encore moins structuré. Une implémentation java est disponible par l'Inria, cependant lors de la recherche d'un protocole peer-to-peer, ce dernier était encore en cours de développement ce qui rendait son utilisation très difficile, de plus, la pauvreté de documentation concernant le code ne facilitait pas la tâche d'apprentissage. C'est pour cela que nous n'avons pas retenu *Gossip* pour le réseau peer-to-peer

2.1.2 Jxta

JXTA est une technologie open source permettant de faire un réseau peer-to-peer en Java. Le principe est simple, un réseau peer-to-peer est créé sur le réseau existant permettant aux différents noeuds de pouvoir communiquer entre eux.

Étant un projet connu, la communauté derrière cette technologie et l'abondante documentation ont permis de choisir et d'utiliser *Jxta* assez facilement.

Chapitre 3

Conception

Chapitre 4

Réalisation

Conclusion

Bibliographie

- [1] Apache. *Maven Documentation*.
- [2] Eric Bidelman. Stream Updates With Server-Sent Events.
<http://www.html5rocks.com/en/tutorials/eventsource/basics/>, November 2010.
- [3] Yann Busnel. Structuring Overlays With Gossip.
- [4] Stéphane Weiss. *Edition collaborative massive sur réseaux Pair-à-Pair*. PhD thesis, Université Henri Poincaré – Nancy, 2010.