

UV « Projet » – Mise-en-oeuvre d'un reseau
logique intégrant la notion de distance
(initiative open-source **DisCoVEry**)

30 septembre 2013

Tuteurs	Marin Bertier – Cédric Tedeschi – Adrien Lebre – Jonathan Pastor
Projet	DisCoVEry de Vivaldi– (Mines de Nantes)
Site Web	http://beyondtheclouds.github.io
Intitulé	Mise-en-oeuvre d'un reseau logique reposant sur l'agorithme Vivaldi

Domaines

- Intégration logicielle
- Informatique Utilitaire (Cloud Computing)
- Langages à objets
- Modèle à base d'acteurs
- Algorithmique distribuée (réseaux logiques)

Compétences requises

- Pratique de la programmation (Java/Scala)
- Maîtrise de principes de programmation objets
- Outil de développement : Eclipse/IntelliJ

Compétences à acquérir

- Développement d'un réseau logique de type vivaldi
- Développement en environnement contraint (intégration des développements dans un contexte logiciel existant)
- Développement en mode agile
- Contribution à un projet « open source »

Contexte :

Afin de supporter la demande croissante d'informatique utilitaire (UC) tout en prenant en compte les aspects énergétique et financier, la tendance actuelle consiste à construire des centres de données (ou centrales numériques) de plus en plus grands dans un nombre limité de lieux stratégiques. Cette approche permet sans aucun doute de satisfaire la demande tout en conservant une approche centralisée de la gestion de ces ressources mais elle reste loin de pouvoir fournir des infrastructures de calcul utilitaire efficaces et durables. L'initiative de recherche DisCoVEry propose une approche radicalement différente permettant de prendre en compte la localité des demandes dès le départ. Pour ce faire, elle propose de tirer parti de tous les équipements disponibles sur l'Internet afin de fournir des infrastructures de calcul utilitaire qui permettront de part leur distribution de prendre en compte plus efficacement la dispersion géographique des utilisateurs et leur demande toujours croissante [1]. Un des aspects critique pour l'émergence de telles plateformes de calcul utilitaire "local" (LUC) est la disponibilité de mécanismes de gestion appropriés exploitant les dernières contributions en terme d'algorithmiques distribués. En particulier, il est primordial d'être capable d'estimer la distance entre les noeuds d'une plate-forme à large échelle.

Le problème : distance et localité entre les noeuds

L'algorithme Vivaldi [2] est un algorithme distribué qui permet de donner à un ensemble de noeuds interconnectés des coordonnées dans un plan en

fonction de leur éloignement physique. Vivaldi s'appuie sur des communications deux à deux, au cours desquelles les noeuds communicants estiment leur RTT (round trip time). A partir de la valeur observée, ils appliquent un calcul simple les rapprochant ou les éloignant dans le plan. En répétant ce processus sur des noeuds choisis aléatoirement dans le réseau, les noeuds convergent rapidement vers leur position finale dans le plan. Cette position peut ensuite être exploitée pour découvrir des noeuds proches physiquement. En effet, des noeuds proches dans le plan le seront aussi physiquement.

La notion de distance fournie par Vivaldi permet de construire des réseaux logiques où la notion de distance physique entre les noeuds est importante. On souhaite ainsi construire une topologie logique reflétant les distances physiques entre les noeuds reliés. Il est en particulier nécessaire d'être capable, depuis n'importe quel noeud du réseau, de retrouver les noeuds les plus proches physiquement afin de les interconnecter. Une façon de réaliser cela est de s'appuyer sur une version modifiée de l'algorithme de Dijkstra pour le calcul du plus court chemin.

Les objectifs

La problématique du projet est l'implémentation de l'algorithme Vivaldi dans un contexte logiciel existant, et son expérimentation sur une plate-forme réelle telle que la plate-forme Grid'5000 (<http://www.grid5000.fr>).

Le projet sera validé au travers quatre objectifs successifs :

- Mise en œuvre du réseau logique Vivaldi en suivant un modèle de programmation de type acteur
- Mise en œuvre d'une API permettant d'exploiter cette notion de distance depuis les mécanismes de plus haut niveau.
- Mise en œuvre d'un mécanisme permettant le parcours du réseau logique de manière efficace (i.e., sur une notion de plus court chemin) en s'appuyant sur l'API bas niveau.
- Validation du mécanisme de parcours au sein de la proposition DVMS.

Les livrables :

- L0 : Cahier des charges (rappel du contexte et des enjeux, formulation des besoins en développements logiciels)
- L1 : Rédaction du cahier d'analyse et conception (Définition d'une feuille de route, des solutions logicielles retenues et les différentes étapes qui permettront de valider la bonne avancée du projet).)
- L2 : implémentation de la solution en tant qu'OverlayActor et validation unitaire.
- L3 : Implémentation de l'API
- L4 : Implémentation du mécanisme de parcours

- L5 : Validation de l'intégration dans le microcosme logicielle DisCoVEry (étude de la pertinence dans le mécanisme DVMS [3]).
- L6 : synthèse des méthodes et techniques mises en œuvre

Méthodologie du projet :

Ce projet sera réalisé suivant une méthode agile. Il devra impliquer une contribution au projet « open source » DisCoVEry.

Bibliographie :

- [1] Adrien Lebre, Jonathan Pastor, Marin Bertier, Frédéric Desprez, Jonathan Rouzaud-Cornabas, Cédric Tedeschi, Paolo Anedda, Gianluigi Zanetti, Ramon Nou, Toni Cortes, Etienne Rivière, and Thomas Ropars. Beyond The Cloud, How Should Next Generation Utility Computing Infrastructures Be Designed? Research Report RR-8348, Inria, July 2013.
- [2] Frank Dabek, Russ Cox, M. Frans Kaashoek, Robert Morris : Vivaldi : a decentralized network coordinate system. SIGCOMM 2004
- [3] Flavien Quesnel, Adrien Lèbre, and Mario Südholt. Cooperative and Reactive Scheduling in Large-Scale Virtualized Platforms with DVMS. Concurrency and Computation : Practice and Experience, December 2012.