

SMA : Projet EDF

Yoann Estepa et Jean Ogier du Terrail

7 janvier 2015

Introduction

Il s'agit dans ce projet de modéliser le comportement de tous les acteurs du marché de l'électricité. A première vue il en existe deux grandes catégories : les clients et les producteurs, mais nous verront que ce modèle même très simple nécessite plusieurs modifications. Il sera aussi l'occasion d'utiliser le framework Jade qui contient toute la structure d'un modèle multi-agents simple ainsi que de tester le logiciel de versioning Git très populaire chez les développeurs.

1 Les Agents et leur contenu

Tous les agents possèdent une méthode `setup` dans laquelle sont codés des Behaviours qui peuvent être cycliques (traitement identique à chaque prise de contrôle de l'agent), one-shot (comportement effectué une unique fois), temporels (au bout d'un timer ou périodiquement) ou encore génériques et conçus par l'utilisateur. En définissant ces comportements nous déterminons complètement le comportement de ces agents et Jade se chargera du reste. Pour l'instant nous avons définis ensemble une structure de conversations et d'interactions entre nos agents sans réellement qu'il y ait de véritable transaction d'argent et de problèmes de dettes ou d'impayés afin de proposer un premier modèle simple qui sera enrichi plus tard.

1.1 ClientAgent

Ce client a besoin de consommer en moyenne `meanProduction` mais sa consommation dépend beaucoup de ses activités qui ne sont pas toujours les mêmes. A chaque tick d'horloge chaque consommateur tire donc une quantité d'électricité à consommer selon une gaussienne centrée de variance définie. Cette action est définie dans un `TickerBehaviour` afin d'être exécutée périodiquement (modélisation d'une consommation mensuelle). Pour trouver cette électricité cet agent va souscrire un abonnement auprès d'un Producteur. Cette action n'est pour l'instant effectuée qu'à la création de l'agent mais pourra être périodisée pour changer d'abonnement en cas de baisse des prix ou l'apparition de nouveaux producteurs.

Tout d'abord, le client contacte le service de DF pour prendre connaissance de la totalité des producteurs d'électricité disponibles. Cette action est réalisée en boucle jusqu'à l'apparition d'au moins un producteur référencé. Il demande les prix des producteurs disponibles (envoi d'un CFP) et reçoit des propositions (voir fournisseur). Comme il est intelligent il prend le moins cher et s'abonne quand il a complété son dialogue¹. Chaque Client possède en attribut un producteur qu'il initialise avec les bonnes valeurs quand la proposition est acceptée. Une fois qu'il est abonné il reçoit tous les mois (voir horloge) des REQUESTS pour effectuer la transaction électricité contre argent.

1.2 Le fournisseur

Le fournisseur possède un prix de vente au kilo qui est public et que tous les clients peuvent demander. Ils ont aussi, heureusement pour eux, une liste de clients qui se remplit au fur et à mesure des souscriptions. Le fournisseur s'inscrit dans le DF à sa création (OneShotBehaviour) pour se faire connaître. Quand il reçoit un "Call for Proposal" (CFP) de la part d'un client, le producteur PROPOSE son prix aux clients qui peuvent faire le choix de l'accepter ou non. Justement le producteur attend les performatifs REFUSE ou ACCEPTE. Dans le cas où sa proposition est acceptée il INFORME le client qu'il est bien abonné et il ajoute ce dernier dans sa liste personnelle. Pour collecter l'argent de ses clients il envoie tous les mois des REQUESTS à l'ensemble de sa liste de clients.

1.3 Le dernier agent : l'horloge

Pour rappeler aux autres agents la notion du temps qui passe nous avons définis un agent horloge possédant un comportement périodique (TickerBehaviour). Cet agent INFORME seulement tous les producteurs qu'il est temps de récolter leur dû et donc d'envoyer les REQUESTS envers leurs clients respectifs. Ces messages aux producteurs se différencient de ceux des clients par l'identifiant de conversation "top". Il a été utile de créer des messageTemplate pour différencier les messages et ne pas dépiler automatiquement des messages relevant d'une autre conversation (ie. considérer un CFP dans le comportement qui gère les tops d'horloge et entraîne la perte du message). Cela a aussi justifié l'utilisation d'ID de conversations.

1.4 L'interface Homme-Machine

Nous avons prévu deux HMI pour gérer notre simulation : une interface graphique et un ensemble de logs permettant de faciliter le suivi des communications. Nous avons commencé à travailler sur une interface graphique sommaire comportant quelques boutons et affichages(producteurs, Nouveau Producteur, Nouveau Client). Notre but est de la faire afficher la liste des producteurs existants sur le marché et de pouvoir les sélectionner pour afficher leurs détails. Dans cette deuxième fenêtre on souhaite avoir la liste des clients qui lui sont liés

1. voir le diagramme de séquence pour l'abonnement en annexe

ainsi qu'un ensemble de données (CA, production mensuelle etc...). Nous souhaitons aussi ajouter la possibilité de rajouter par interface graphique de nouveaux agents en cours de simulation mais JADE propose déjà un tel outil. Par la suite, il peut être intéressant de la faire évoluer en lui rajoutant les messages consoles voire de lui donner un aspect plus ludique (barres de consommation d'électricité, état des comptes clients, etc.).

Conclusion

Après avoir passé beaucoup de temps à comprendre le système de Jade et à mettre en place le processus d'abonnement, il est maintenant plus facile de complexifier les comportements et de rajouter un côté financier à notre marché jusque là fondé sur la confiance.

Annexes

Diagrammes de séquence

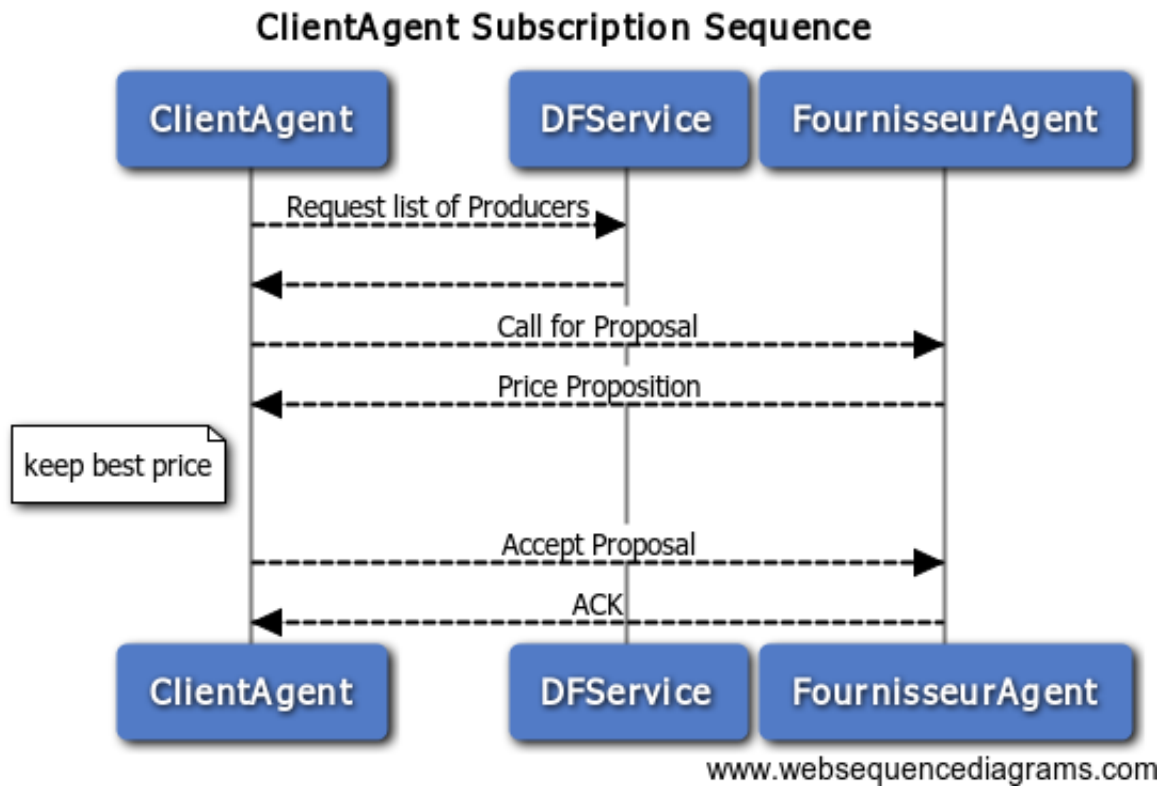


FIGURE 1 – Diagramme de séquence de souscription d'un Client auprès d'un Fournisseur

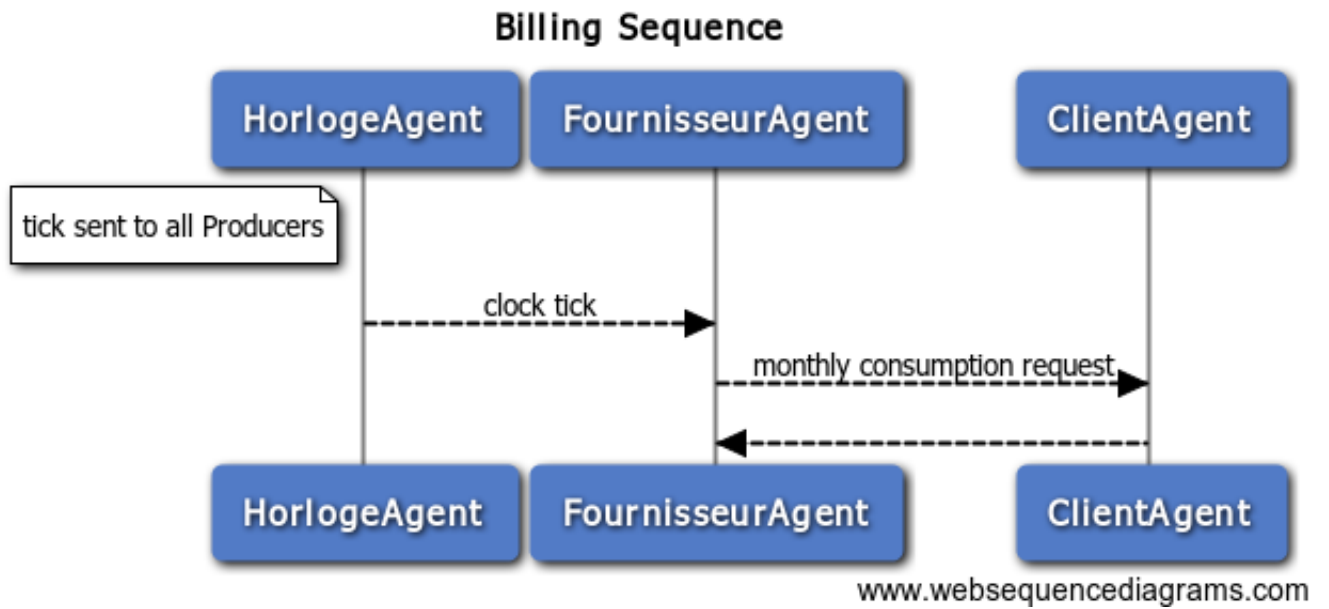


FIGURE 2 – Diagramme de séquence de facturation d'un Client