

**"Développement de Web Services,
mise en oeuvre appliquée à deux plates-formes INRA : MEANS et RECORD"**

- Stage effectué par Rabah Meradi, étudiant en Licence Informatique à l'Université Paul Sabatier.
- Encadrantes : Nathalie Rousse (MIAT, RECORD) et Julie Auberger (SAS, MEANS).
- Période du 7 avril au 11 juillet 2014.

Introduction

Ce stage a été effectué par Rabah Meradi, étudiant en Licence Informatique à l'Université Paul Sabatier. Les encadrantes étaient Nathalie Rousse (MIAT, RECORD) et Julie Auberger (SAS, MEANS). Le stage s'est déroulé du 7 avril au 11 juillet 2014.

L'utilisateur ('client') considéré dans le cadre du stage est la plateforme MEANS, dédiée à l'analyse multicritère de la durabilité des processus de production agricole et de transformation des produits issus de l'agriculture. Cette plate-forme s'appuie sur les sorties de divers modèles de calcul parmi lesquels on aimerait pouvoir compter des modèles de la plate-forme RECORD. Le modèle particulier considéré dans le cadre du stage est le système de culture MicMac.

Le stage a permis d'illustrer la faisabilité de communication par services web entre la plate-forme MEANS et la plate-forme RECORD, créant un pont entre d'un côté la simulation de systèmes de production (RECORD) et de l'autre le calcul d'indicateurs d'impact (MEANS).

Présentation

L'objectif du stage était de mettre en place des services web permettant à la plate-forme MEANS de piloter la simulation du système de culture MicMac, à partir de laquelle l'utilisateur de la plate-forme MEANS pourrait ensuite calculer des indicateurs d'impact (pour plus d'informations : le [sujet du stage](#)).

Il s'agissait d'une part de développer (en python) les traitements de services web et d'autre part de développer (en java) l'API qui permet d'appeler ces services web depuis le logiciel de la plate-forme MEANS. La version utilisée du modèle MicMac est celle programmée sous la version vle-1.0.

Schéma de l'architecture logicielle

Les informations échangées sous forme de services web par les plates-formes MEANS et RECORD correspondent au cas d'utilisation de la plate-forme MEANS (les traitements ont été développés autour du modèle MicMac de la plate-forme RECORD vis-à-vis des besoins spécifiques de la plate-forme MEANS). Pour plus d'informations : voir la documentation en ligne sur <http://147.100.164.29/> (adresse de mise en ligne des services web) (*adresse obsolète, ces services web ne sont plus en ligne*).

Productions

- **Services web** : les services web produits sont en ligne à l'adresse <http://147.100.164.29/>, l'appel de ces services web est d'accès limité à l'équipe MEANS (*adresse obsolète, ces services web ne sont plus en ligne*).
- **Code source** : le code source de la production logicielle (services web et API) peut être téléchargé en allant sur la forge [Mulcyber](https://mulcyber.toulouse.inra.fr/projects/recordweb), dans le dépôt SVN du projet [RECORD WEB](#) (accès public) (adresse 'https://mulcyber.toulouse.inra.fr/projects/recordweb'). Les tags de livraison se situent sous 'tags/meansrecordws' pour le logiciel des services web et sous 'tags/meansrecordapi' pour le logiciel de l'API, la branche de développement relative au stage est 'branches/030_ws_means'.

Appel des services web

Exemples d'appel de services web développés dans le cadre du stage :

Rappel : l'appel de ces services web est d'accès limité à l'équipe MEANS (il faut un compte 'user' et 'mot de passe').

Ces services web ne sont plus en ligne. Les liens qui suivent sont obsolètes.

Utilisation des services web depuis le navigateur

1) Aller sur <http://147.100.164.29/rwtool/admin/> pour s'authentifier (user 'xxx', mot de passe 'yyyy').

2) Lancement d'une simulation :

Aller sur <http://147.100.164.29/simulation/MicMac/>

Copier les paramètres d'appel dans la zone de texte (**Content**) :

Exemple :

- en 'Media type' valant 'application/x-www-form-urlencoded', copier :

culturePrincipale=BleDur&culturePrecedente=Tournesol®ion=Auzeville

- en 'Media type' valant 'application/json', copier :

```
{ "culturePrincipale" : "BleDur", "culturePrecedente" : "Tournesol", "region" :  
"Auzeville" }
```

Voir [illustration](#)

Puis cliquer sur POST, afin de recevoir les résultats de la simulation de MicMac :

Voir [illustration](#)

Un exemple d'appel en ligne de commandes avec 'curl'

1) Obtention du token :

La commande

```
curl -X POST -H "Content-Type: application/json" -d
```

```
'{"username": "xxx", "password": "yyyy"}' http://147.100.164.29/api-token-auth/
```

retourne le token :

```
{"token":
```

```
"eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VybmFtZSI6ImRlbW8iLCJ1c2VyX2lkIjo0LCJlbWFpbCI6IiIsImV4cCI6MTQwNDg5OTU3Mn0.2BWYGkOZxDq7inpTBfYdB8nT_G2M4bzmWfG6dtXdijU" }
```

2) Lancement d'une simulation :

Commande (où il faut remplacer le token par celui obtenu à la commande précédente) :

```
curl -X POST -H "Content-Type: application/json" -H "Authorization: JWT
```

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VybmFtZSI6ImRlbW8iLCJ1c2VyX2lkIjo0LCJlbWFpbCI6IiIsImV4cCI6MTQwNDg5OTU3Mn0.2BWYGkOZxDq7inpTBfYdB8nT_G2M4bzmWfG6dtXdijU" -d '{"culturePrecedente": "Tournesol", "culturePrincipale": "BleDur", "region": "Auzeville"}' http://147.100.164.29/simulation/MicMac/
```

Appel depuis un programme Java

Voir le test d'appel inclu dans le code de l'API java produite dans le cadre du stage.

Voir aussi le [manuel d'utilisation de l'API java](#) permettant à la plate-forme MEANS d'appeler des services web de la plate-forme RECORD.

Appel depuis un programme Python

Voir le test d'appel inclu dans le code des web services produits dans le cadre du stage.

Un exemple d'appel sous R

```
library('RCurl')  
library('rjson')
```

1) Obtention du token :

```
header = c('Content-Type' = 'application/json', Accept = 'application/json')  
postfields = toJSON(list(username = "xxx", password = "yyyy"))  
restoken = postForm( uri="http://147.100.164.29/api-token-auth/", .opts = list(postfields =  
postfields, httpheader = header))  
TOKENVALUE = fromJSON(restoken)$token
```

2) Lancement d'une simulation :

```
token = paste("JWT", TOKENVALUE, sep=" ")  
header = c('Content-Type' = 'application/json', Accept = 'application/json', 'Authorization' =  
token )  
postfields = toJSON(list(culturePrecedente = "Tournesol", culturePrincipale = "BleDur",  
region = "Auzeville"))  
ressimu = postForm( uri="http://147.100.164.29/simulation/MicMac/", .opts = list(postfields  
= postfields, httpheader = header))  
RESSIMU = fromJSON(ressimu)
```