

Galaxy4Bioinformatics

Développement et intégration d'application sous 



Olivia Doppelt-Azeroual
Fabien Mareuil
Alban Lermine



5 Novembre 2014

TP: Bioblend l'API de Galaxy



Pour commencer, démarrez ipython et connectez vous à votre galaxy avec bioblend

Récupérez votre api key et l'url de galaxy

- **Chargez le module bioblend :**

- `from bioblend.galaxy import GalaxyInstance`

- **Créez un objet GalaxyInstance avec votre url et votre key**

- `gi = GalaxyInstance(url="http://...", key="votre key")`

- **ipython: la complétion automatique**

- écrivez gi. puis appuyez sur la touche tab
- Vous pouvez ensuite vous aidez de help(command) pour comprendre les méthodes/classes
- Utilisez le module pprint pour mieux afficher les dictionnaires json

Familiarisez vous avec les classes et les méthodes du package bioblend (<http://bioblend.readthedocs.org/en/latest/>)

ipython peut vous aider à explorer toutes ces classes et méthodes, abusez en.

- **Les historiques :**

- `get_histories`, `create_history`, `show_history`, `show_dataset`,
`download_dataset`, `download_history`..

- Essayer de récupérer la liste entière des données d'un historique
- Créez un nouvel historique

- **Les librairies :**

- Faites pareil avec les `libraries`, `get_libraries`, `show_library`..
- Créez une `library` puis importez des données dedans

- Copiez des données de votre `library` dans un nouveau `history`

Familiarisez vous avec les classes et les méthodes du package bioblend (<http://bioblend.readthedocs.org/en/latest/>)

ipython peut vous aider à explorer toutes ces classes et méthodes, abusez en.

- **Les outils :**

- `from bioblend.galaxy.tools.inputs import inputs, dataset`
- Retrouvez puis exécutez votre outil
- Récupérez le/les résultats, `histories.download_dataset...`

- **Les workflows :**

- `workflows.get_workflows()`, `show_workflow`, `run_workflow`
- (si ce n'est pas déjà fait) Créez un petit workflow dans galaxy
- Récupérez et exécutez votre workflow avec l'api



Une pause? pourquoi faire...

Challenge :

- Écrivez un script permettant d'exécuter votre workflow avec vos données.



Des commandes qui peuvent vous aider:

La librairie :

```
gi.libraries.get_libraries()
gi.libraries.get_libraries(name='malibrairie')
gi.libraries.show_library('library_id')
gi.libraries.show_library('library_id', content=True)
gi.libraries.create_library('nom')
gi.libraries.set_library_permissions('library_id', access_in=['user_id'],
modify_in=['user_id'], add_in=['user_id'], manage_in=['user_id'])
gi.libraries.upload_file_from_local_path('library_id', file_local_path)
```

L'historique :

```
gi.histories.get_histories()
gi.histories.get_histories(name='monhistorique')
gi.histories.get_current_history()
gi.histories.show_history('history_id')
gi.histories.show_history('history_id', contents=True)
gi.histories.create_history('nom')
gi.histories.upload_dataset_from_library('history_id', 'lib_dataset_id')
```

Des commandes qui peuvent vous aider :

Les outils (fonctionnel avec les dernières versions de bioblend, voir github) :

```
from bioblend.galaxy.tools.inputs import inputs, dataset
gi.tools.get_tools(name="monoutil")
gi.tools.show_tool('tool_id')
myinputs=          inputs().set("inputName", 'value').set("input",dataset
('dataset_id'))
gi.tools.run_tool('history_id','tool_id', tool_inputs=myinputs)
```

Récupérez le/les résultats :

```
gi.histories.show_history('history_id')
gi.histories.download_dataset('history_id', 'dataset_id')
```

Attention : Les dataset_id ne sont pas uniques

Des commandes qui peuvent vous aider :

Les workflows :

Récupération du workflow id et de l'identifiant du dataset (dataset_map):

```
gi.workflows.get_workflows()  
gi.workflows.import_workflow_from_local_path("./my_workflow.ga")  
workflow = gi.workflows.show_workflow('workflow_id')  
dataset_map={workflow["inputs"].keys()[0]: {"id": 'dataset_id', "src":  
["hda", "ld", "llda"]}}  
gi.workflows.run_workflow('workflow_id', 'history_id',  
dataset_map=dataset_map)
```

Les status d'un history ou d'un workflow :

```
gi.histories.get_status('history_id')
```




Quelques connaissances acquises dans la douleur

Pré-requis :

- authentification externe (LDAP)
- nouvel utilisateur tout frais (vous possédez son mot de passe)

Besoin :

- créer des bibliothèques automatiquement (intègre un pipeline de création de nouvel utilisateur sur la plate-forme)

Problème :

- on ne peut pas créer de bibliothèque pour un utilisateur qui ne s'est jamais connecté

Solution proposée :

- établir un premier contact avec lynx

```
cmd="lynx -dump -auth=\"%s:%s\" %s > /dev/null" % (user,user_password,url)
```



Quelques connaissances acquises dans la douleur

Problème :

- Faire cohabiter l'authentification externe (LDAP) avec l'utilisation de l'API
- Votre apache ne permettra le passage de requêtes que si vous avez rentré votre login mot de passe, les requêtes via l'api en ligne de commande n'intègrent pas le login, mot de passe (enfin il ne vaut mieux pas)

Solution proposée :

- Démarrez 2 serveurs Galaxy :
 - le premier configuré normalement pour l'utilisation via browser de l'instance galaxy avec une authentification externe (LDAP)
 - le deuxième ne permettant ni la création d'utilisateur ni l'authentification anonyme, mais sans authentification externe (LDAP) (ouvert à tous)
- Vous pourrez alors utiliser le serveur "api" pour votre utilisation de l'api en ligne de commande

Quelques connaissances acquises dans la douleur

Problème :

- La méthode actuelle de `download_dataset` ne fonctionnera pas si vous êtes en `https` ou si vous avez choisi l'option `require_login = True` dans la configuration de `galaxy`

Solution :

- Une nouvelle version de la méthode `download_dataset` est en cours de réflexion dans l'équipe qui développe `bioblend`

Galaxy4Bioinformatics

Développement et intégration d'application sous 



Olivia Doppelt-Azeroual
Fabien Mareuil
Alban Lermine



5 Novembre 2014

MERCI de votre Attention

