Université Bordeaux 1
Master 2 Bioinformatique

Analyse technique de logiciel
Galaxy

Florian CARRE
Benjamin DARTIGUES
Sebastien BEAUQUIS
Guillaume BERNARD
Tom LESLUYES

Responsable : Mme. THEBAULT

Bordeaux, le 30 Janvier 2012

Table des matières

Introduction **1**

1 Présentation **2**
1.1 État de l'art . . . 2
1.1.1 CLC bio genomics workbench . . . 2
1.1.2 Genomatix . . . 2
1.1.3 Ergatis . . . 3
1.1.4 Biomoby . . . 4
1.1.5 Lasergene 9.1 . . . 4
1.1.6 JMP Genomics . . . 4
1.2 Séquençage nouvelle génération . . . 5

2 Utilisation **9**
2.1 En ligne . . . 9
2.2 En local . . . 10
2.2.1 Téléchargement du code source . . . 11
2.2.2 Exécution du serveur local . . . 12

3 Fonctionnalités **14**
3.1 Présentation générale . . . 14
3.2 Ajout de plug-ins . . . 15
3.2.1 Création . . . 15
3.2.2 Intégration . . . 15
3.3 Workflow . . . 17

Conclusion **19**

Introduction

Le séquençage du génome, c'est à dire l'ensemble du code ADN d'un être vivant, s'est transformé ces dernières années en une course effrénée entre laboratoires concurrents.

Cette course a contribué à l'explosion de la quantité de données à traiter, données fournies par les séquenceurs nouvelle génération. L'étude de ces données nécessite de posséder des outils de grande fiabilité mais surtout possédant un large panel de traitements applicables à ses mêmes données.

En effet, étudier le génome ne se limite pas à un simple déchiffrage du code contenu dans l'ADN, l'objectif reste principalement de comprendre le génome en analysant les gènes et les régions intergéniques, en évaluant le niveau d'expression de gènes par rapport à d'autres, en comparant les génomes de différentes espèces entre eux afin d'en sortir des lois et des règles.

Il existe de nombreuses plates formes sur le web ainsi que de nombreux logiciels dédiés à l'étude des données issues du séquençage. Notre regard s'est porté sur Galaxy [1], à la fois une plate forme web et un logiciel open source, qui présente une multitude de traitements à appliquer aux données du génome.

Nous présenterons ainsi une revue sommaire des logiciels existants à ce jour puis nous vous exposerons une série de traitements réalisés afin de mettre en exergue les qualités et les défauts inhérents à ce logiciel.

1. [1][2][3][4][6][7][8][9][10][13][15][16][17][18][19]

1 Présentation

1.1 État de l'art

Nous présentons ici une revue des principaux logiciels "concurrents" ou du moins répondant aux mêmes types de problématiques que Galaxy. Certains parmi eux ne réalisent qu'une petite partie des traitements accomplis par Galaxy. Quant aux autres, rares sont ceux qui offrent autant de liberté et de facilité d'utilisation. De plus, la plupart sont payants et offre donc seulement des versions d'essais limités.

1.1.1 CLC bio genomics workbench

Description

C'est une plate forme commerciale et une extension du logiciel CLCbio Main Workbench . On peut uniquement obtenir une version d'essai pour 15 jours.

- assembleur de séquences de novo (inclus la détection des sNP , CHiP-seq).
- Composition du génome.
- Analyse des séquences (gènes et régions intergéniques).
- Nombre de gènes et leur position sur les chromosomes.
- Niveau d'expression des gènes.
- Comparaison des génomes de diverses espèces (génomique comparative).

Système d'exploitation

Windows, Mac OS X et Linux.

1.1.2 Genomatix

Description

Genomatix propose des solutions et des services pour l'ensemble du déroulement de l'analyse, de la cartographie de premier niveau à l'intégration des résultats multiples avec des copies des fond des données de haute qualité. Les technologies de visualisation et d'interprétation permettent aux scientifiques du monde entier de transformer leurs données en résultats significatifs, transformant le séquençage nouvelle génération en un outil parfait appliqué à la médecine personnalisée. Il se décompose en trois grandes entités que sont Genomatix Genome Analyzer (GGA), Genomatix Station mining (GMS) et Genomatix Software Suite.

Genomatix Genome Analyzer (GGA)

Genomatix Genome Analyzer (GGA) est une solution intégrée complète pour la visualisation et l'interprétation de Next Generation Sequencing (NGS) de données de la puce à ARN, d'ADN,ou de petit séquençage d'ARN. Chaque analyseur est pourvu d'un état de l'art des technologies ce qui éclaire le contexte biologique. Le GGA produit des résultats de grande pertinence.
Les données de base biologiques comprenant des données du réseau d'annotation et de gènes fournis par Eldorado, ainsi que la connaissance des facteurs de transcription contenues dans MatBase permettent aux chercheurs d'analyser et d'interpréter leurs résultats expérimentaux dans un contexte biologique unique sur chaque GGA et sur plus de 30 espèces différentes. L'analyse de l'expression différentielle (jusqu'au niveau de transcription), la vérification du réseau, l' analyse de la littérature ne sont que quelques-unes des tâches qui peuvent être effectuées.

Genomatix Station mining (GMS)

Genomatix Station Mining (GMS) offre un alignement haute performance des séquençages de nouvelle génération (NGS). Elle permet de lire sur les génomes, transcriptomes, petit ARN. Avec son interface utilisateur intuitive, le GMS vous aide à exécuter rapidement des tâches telles que le positionnement génomique, SNP et détection d'Indel, des analyses d'épissage, la fusion de gènes et l'analyse structurelle. De plus, la station reste souple tout au long de l'expérience. En combinaison avec le Genome Analyzer Genomatix on obtient une solution d'analyse entièrement intégrée à partir du séquençage et pratique pour l'interprétation.

Genomatix Software Suite

Un bundle de logiciels bien établis, la suite logicielle Genomatix effectue un certain nombre de tâches :

- elle procède à une analyse scientifique des données génomiques, la régulation des gènes et l'expression,
- elle génère et évalue les réseaux et les voies,
- elle effectue des recherches documentaires étendues et des analyses de séquence et de l'extraction,
- elle visualise les annotations des génomes complets.

1.1.3 Ergatis

Description

Ergatis est un utilitaire-Web utilisé pour créer, exécuter et contrôler les pipelines réutilisables en analyse informatique. Il contient des composants pré-construits pour les tâches courantes d'analyse bioinformatique. Ces composants peuvent être disposées graphiquement pour former des pipelines hautement configurables. Chaque composant d'analyse soutient plusieurs formats de sortie, y compris le Systems Bioinformatic Markup Language (SBML) qui est un format ouvert XML pour l'échange de séquences et de leurs méta-données. L'implémentation actuelle inclut le support pour le chargement des données dans des bases de projet suivant le schéma Chado (le design d'une base de données particulière), un schéma normalisé soutenue par la communauté pour le stockage des données biologiques.

Ergatis utilise le moteur de workflow pour traiter ses travaux sur une grille de calcul. Workflow fournit un langage XML et du moteur de traitement pour préciser les étapes d'un pipeline de calcul. Il fournit l'état d'exécution détaillé, facilite la récupération d'erreur au point de défaillance, et est hautement évolutive avec un support pour les environnements informatiques les

plus distribués. Le format XML utilisé permet d'exécuter les commandes en série, en parallèle, et dans n'importe quelle combinaison ou niveau d'imbrication.

Ce cadre de travail a été employé dans l'annotation de plusieurs grands organismes eucaryotes, y compris *Aedes aegypti* et *Trichomonas vaginalis*. Ce projet est à ce jour fonctionnel tout en étant en cours de développement actif, avec la plupart des codes proviennent de contribution avec l'Institut des sciences du génome et le J. Craig Venter Institute.

1.1.4 Biomoby

Description

BioMoby est un projet Open Source de recherche qui vise à générer une architecture pour la découverte et la distribution des données biologiques à travers des services Web. Les données et les services sont décentralisés, mais la disponibilité de ces ressources, et les instructions permettant d'interagir avec eux, sont inscrits dans un emplacement central appelé MOBY centrale.
L'approche actuelle étend les services web en mettant en œuvre un modèle de registre novateur qui permet la recherche et la récupération basée sur l'objet et les hiérarchies de service. Cela permet aux utilisateurs de parcourir des données expansives et disparates où chaque étape possible est présenté sur la base des données de l'objet actuellement en main. Les Objets BioMoby natifs sont de type XML, et constituent à la fois la requête et la réponse d'un Simple Object Access Protocol (SOAP) de transaction.

1.1.5 Lasergene 9.1

Description

Lasergene 9.1 comprend de nouveaux assembleurs de séquences "next-gen", des workflows d'analyse et les séquences, structures et vues d'analyse des protéines intégrées. Cela fait vraiment de Lasergene un des logiciels intégrés les plus fiables du marché soutenant l'analyse de séquence traditionnelle, l'assemblage de séquences next-gen ainsi que leurs analyses, les études d'expression génique, l'analyse d'ARN-Seq et de ChIP-Seq.

Système d'exploitation

Windows, Mac et Linux

1.1.6 JMP Genomics

Description

JMP Genomics est un logiciel de découverte statistique issu de deux références dans les logiciels analytiques : SAS et JMP. Les organismes de recherche utilisent JMP Genomics pour découvrir des modes significatifs dans la génétique à haut débit, l'expression des génes, le nombre de copies et les données protéomiques. L'Interaction graphique dynamique rend facile à explorer les données en relation avec un ensemble complet d' algorithmes avancés de statistiques.

Système d'exploitation

On trouve une version 9 qui est compatible avec Windows et Mac. Les versions antérieures sont quand à elles compatibles aussi avec Linux.

1.2 Séquençage nouvelle génération

Historique

Le séquençage de l'ADN a été inventé dans la fin des années 1970. Deux méthodes ont été développées indépendamment. L'une, basé sur la **synthèse enzymatique** sélective et réalisé par l'équipe de Frederick Sanger en Angleterre. La deuxième, basé sur la **dégradation chimique** sélective a été réalisée par l'équipe de Walter Gilbert aux États-Unis. Tout deux ont été récompensé du prix Nobel de chimie pour cette découverte en 1980.

La technique de séquençage de Walter Gilbert était le prémisse du pyroséquençage. Cette technique est principalement basé sur l'addition d'un seul nucléotide qui est révélé en temps réel par détection de la luminescence.
Au début du XXème siècle, de nouvelles techniques de séquençage couplées aux connaissances en physique, chimie, informatique, nanotechnologie et biotechnologie ont vu le jour, et permettent d'augmenter le débit du séquençage par une parallélisation massive des réactions et par miniaturisation des supports utilisés.
Nous allons décrire deux plateformes de séquençage novatrices qui, aujourd'hui encore, sont les modèles en matière de séquençage haut-débit :

- la technologie 454 (le pyroséquençage),
- la technologie Solexa (Illumina).

Les différents types de séquençage haut-débit[14][11][21][5][12]

Le pyroséquençage

Le pyroséquençage est donc une version améliorée de la technique de Maxam et Walter. Il a été élaboré par Hyman et al., un groupe suédois. Le principe est d'utiliser la technique de Walter en l'a couplant avec la technique de PCR (Réaction en Chaine par Polymerase). C'est une technique de séquençage par **synthèse**.

Premièrement, on réalise une bibliothèque des séquences que l'on veut séquencé (à l'aide d'une séquence de ligation qui va se coupler à la séquence).

On extrait la séquence puis on lui fixe une séquence dite adapteur. La séquence adapteur est complémentaire d'une séquence fixée à une micro-bille. On attache donc la séquence d'intérêt à une micro-bille. On dispose de nombreuses micro-billes préparées de la même façon dans des puces à ADN puis on polymérise la séquence par PCR (Fig. 1 et 2).

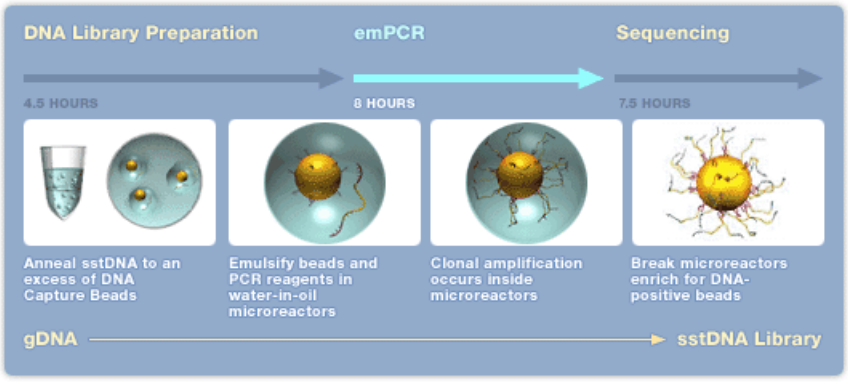

FIGURE 1 – Ligation des séquences aux microbilles.

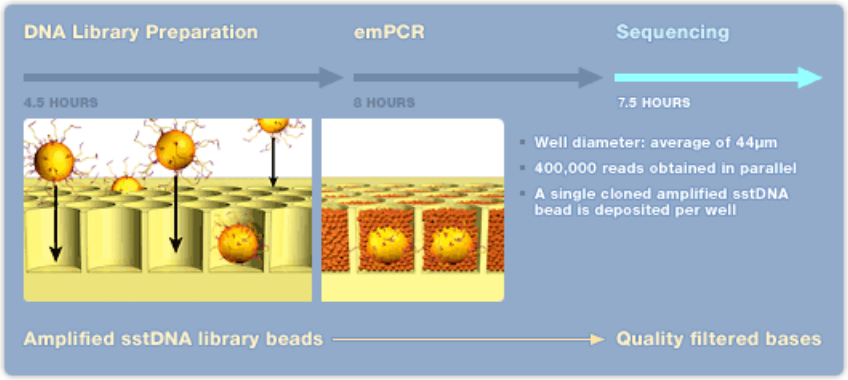

FIGURE 2 – Incorporation des billes dans la puce à ADN.

On dispose la puce dans une cuve contenant :

- de la Sulfurylase,
- de la Luciferase,
- de l'Apyrase,
- de la Polymerase.

Le tableau ci-dessous explique le rôle de chacune des enzymes listées précédemment :

Enzymes	Rôles
Sulfurylase	Convertie un pyrophosphate en adenosine tri-phosphate (ATP).
Luciferase	Emet de la lumière en consommant un ATP.
Apyrase	Dégrade les nucléotides.
Polymerase	Incorpore un acide nucléique complémentaire à un brin d'ADN en consommant un groupement phosphate d'un ATP.

Ensuite, on incorpore successivement les acides nucléiques de manière sélective au niveau de la puce.

A chaque incorporation d'acides nucléiques, le résidu attendu par la polymérase est intégré dans la chaine ADN pendant l'élongation et libère un pyrophosphate.

L'ATPsulfurylase vient alors transformer ce Pyrophophate (PPi) en ATP qui est alors utilisé, couplé à une Luciférine, par une Luciférase. On a alors production d'Oxyluciférine et d'un signal lumineux.

L'Apyrase dégrade les acides nucléotidiques en surplus dans le milieu (Fig. 3).

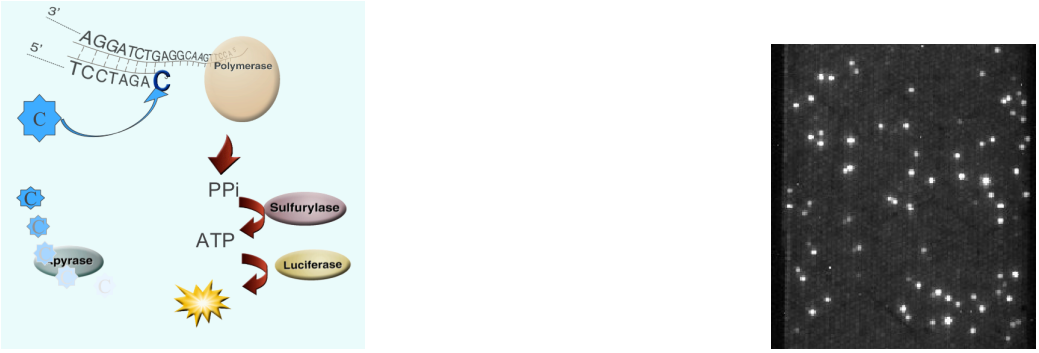

FIGURE 3 – Représentation des étapes réactionnelles pour obtenir le signal lumineux.

Le signale lumineux est capté par un capteur CCD (Charge-Coupled Device) puis reproduit sous forme d'un pic sur le Pyrogramme.La hauteur de ce pic est fonction de l'intensité de la lumière, elle-même proportionnelle de la quantité de l'acide nucléique intégré (Fig. 4).

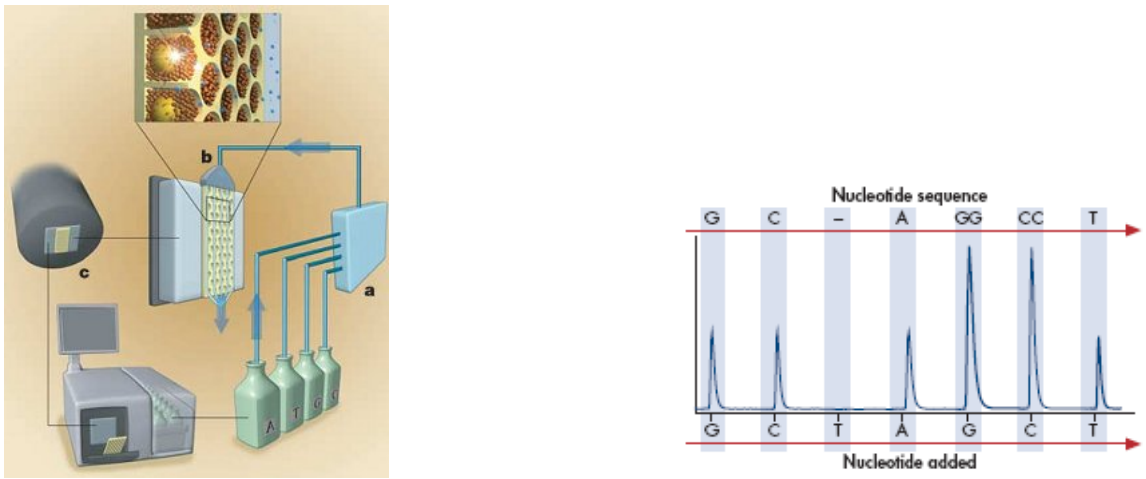

FIGURE 4 – Lecture de la séquence à partir des signaux lumineux.

Technique de séquençage Illumina Illumina est le nom d'un laboratoire développant entre autre le séquençage à partir de la technique de Sanger (la CRT : Cyclic Reversible Termination) ainsi que des connaissances actuelles en biopuces à ADN, nanotechnologies et en informatique de pointe pour l'acquisition, le traitement et l'analyse des images.
Cette technique s'applique en plusieurs étapes :
Étape 1 : Préparation de la banque d'ADN génomique. L'ADN génomique est fragmenté par nébulisation. Les extrémités sont réparées et des adaptateurs sont fixés sur chaque extrémité par ligation.
Étape 2 : Des ponts d'amplification sont formés sur des plaques ce qui permet d'obtenir de grandes quantités de brins d'ADN, augmentant ainsi le débit de séquençage.
Étape 3 : Les brins sont ensuite dénaturés et l'on effectue le séquençage par la technique CRT. Les nucléotides fluorescents sont additionnés d'un groupement nitrophenyl en 3'O. Le nucléotide est incorporé à la séquence lors de la synthèse. Ensuite, il y a une phase de détection du nucléotide inséré par fluorescence. L'envoi d'un rayonnement UV (>300nm) détache le groupement nitrophenil de la liaison en 3'O permettant l'incorporation de nouveaux nucléotides.
Étape 4 : Interprétation des résultats et lecture de la séquence (Fig. 5).

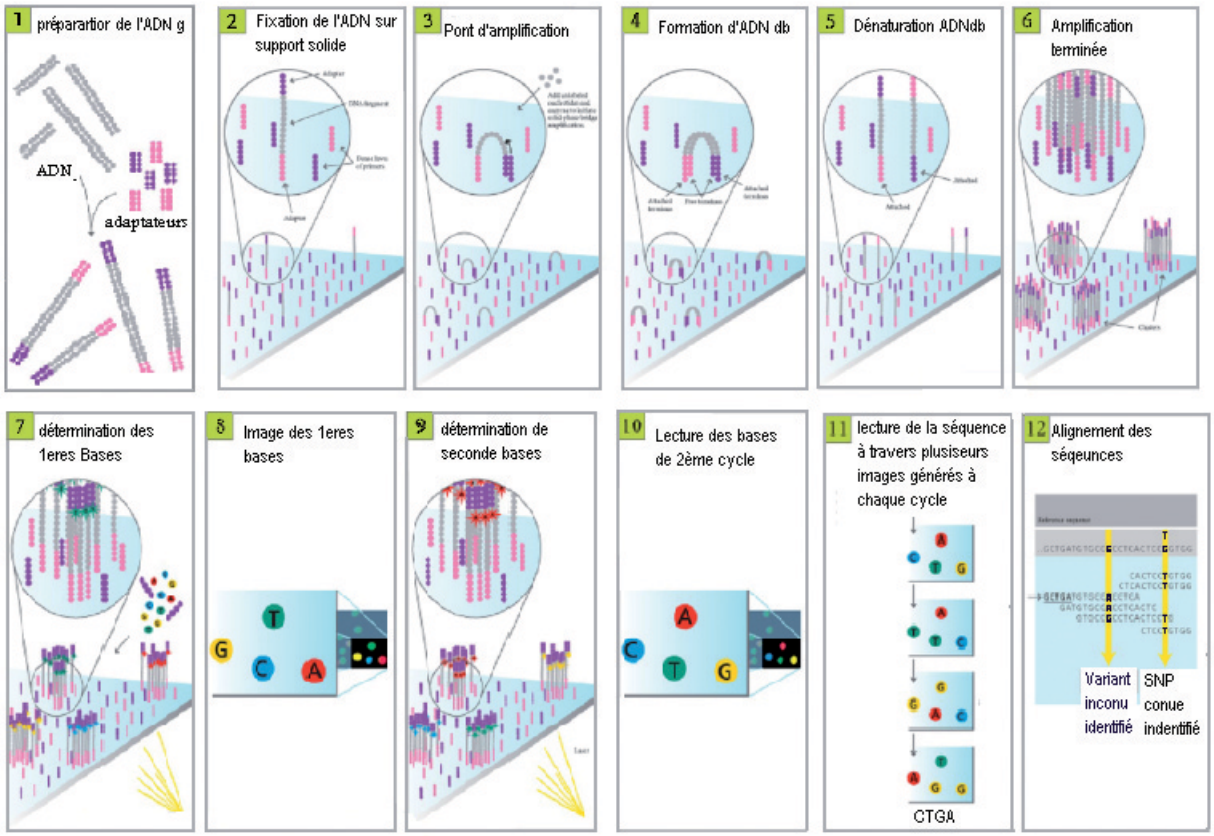

FIGURE 5 – Différentes étapes de la méthode de séquençage Illumina.

2 Utilisation

Galaxy est un logiciel qui peut être utilisé de trois façon différentes :
- en ligne, depuis le site web de ce dernier,
- sur un "cloud",
- en local, directement sur son ordinateur.

La création de sa propre instance de Galaxy sur le "cloud" permet une utilisation de celui-ci similaire à celle en local sans l'inconvénient de la place prise en mémoire sur son ordinateur, toutefois un compte Amazon Web Services est requis.

La création du compte est gratuite, cependant un numéro de carte bleue doit être renseigné, en effet il semble que l'utilisation de l'ensemble des fonctionnalitées du "cloud" ai un coût. Nous n'avons donc pas pu tester Galaxy sur le "cloud".

2.1 En ligne

L'utilisation en ligne de Galaxy est le moyen le plus rapide et le plus accessible : en effet il suffit de se rendre sur le site de ce dernier pour pouvoir y avoir directement accès. Si une utilisation sommaire ne nécessite pas la création d'un compte, il est fortement recommandé d'en créer un (gratuitement) pour avoir accès à plus de fonctionnalité. Posséder un compte sur Galaxy permet :

- d'avoir accès aux données et aux outils d'analyse depuis n'importe quel ordinateur connecté à Internet,
- augmenter les données et les travaux simultanés,
- sauvegarder un historique de compte sur une base systémique,
- d'avoir la capacité de nommer, sauvegarder, partager et publier des objets de Galaxy : des historiques, des workflows, des ensembles de données, des pages.
- de télécharger via le FTP de plus grands ensembles de données.

Galaxy dispose de nombreux outils (plus de 35 différents), heureusement lorsque l'on sélectionne l'un d'entre eux une petite fenêtre explicative nous renseigne directement sur ce qu'il fait, le format de données qu'il accepte et un exemple est également proposé (Fig. 6).

Un wiki dédié à Galaxy est disponible sur son site, plusieurs sections différentes permettent d'en apprendre plus sur les données que peut utiliser Galaxy et les outils qu'il propose. De plus de nombreuse vidéos "tutoriel" sont disponibles, ainsi un exercice nommé Galaxy 101 fait office de "tutoriel" de départ et permet de se familiarisé avec le logiciel et quelques unes de ces fonctionnalitées. Une vidéo "tutoriel" est disponible pour chaque outil intégré à Galaxy (Fig. 7).

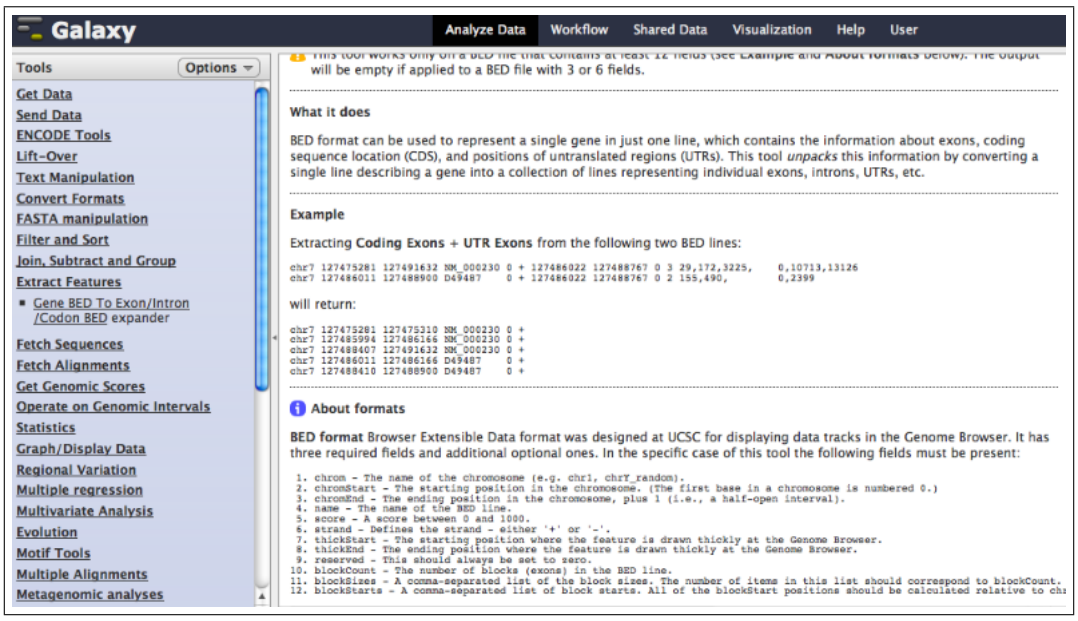

FIGURE 6 – Page d'aide d'un des outils de Galaxy.

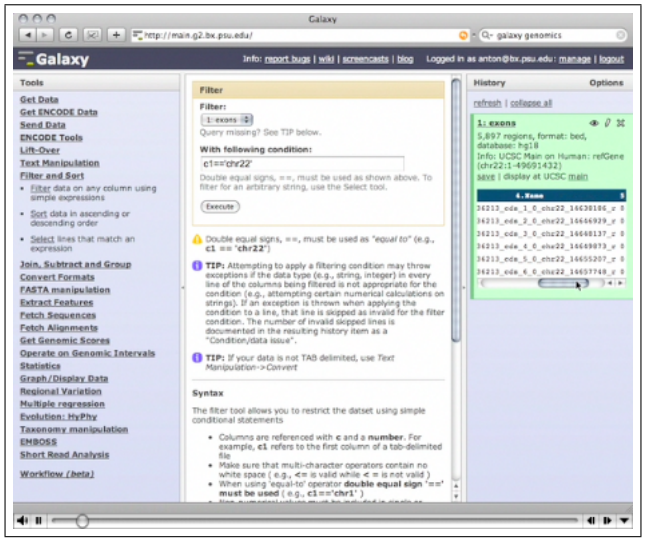

FIGURE 7 – Vidéo "tutoriel" de Galaxy.

2.2 En local

En plus de proposer une utilisation via internet, il est possible d'installer Galaxy sur un ordinateur. L'utilisateur peut ainsi profiter de toutes les fonctionnalités de Galaxy et même les personnaliser en vue d'une utilisation spécifique [1].

Par rapport à l'utilisation en ligne, le local présente plusieurs avantages :
- modifier les paramètres des plug-ins déjà implémentés,
- ajouter ses propres plug-ins,
- augmenter la vitesse d'analyse,
- conserver des données sensibles.

1. Le tutorial officiel de l'installation locale est disponnible à l'adresse suivante : http ://wiki.g2.bx.psu.edu/Admin/Get%20Galaxy

L'installation est très simple et se déroule en deux étapes :

1. téléchargement du code source,
2. exécution du serveur local.

2.2.1 Téléchargement du code source

La dernière version stable de Galaxy est toujours disponible depuis un répertoire Mercurial [2] hébergé sous Bitbucket [3]. Il existe deux façons de récupérer le code source : en utilisant les outils proposés par Bitbucket ou en utilisant Mercurial.

Bitbucket

Bitbucket est un site d'hébergement pour les systèmes de contrôles de versions Git [4] et Mercurial. Il offre aussi de nombreux outils tels que : Issue tracker, Wiki, Basecamp, Flowdock, Twitter, etc ...

Ses principaux avantages sont la facilité d'utilisation, sa gratuité (sous réserve d'avoir moins de cinq collaborateurs sur tous les répertoires privés) ainsi que les aides proposées (nombreux tutoriaux, large communauté, Google group, I.R.C, etc ...).

Le téléchargement de Galaxy est simple : il suffit, dans l'onglet *Source* (Fig. 8), de cliquer sur *get source* et de sélectionner le mode de compression désiré (Fig. 9).

FIGURE 8 – Onglets disponibles (Bitbucket).

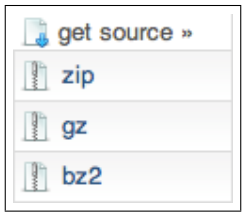

FIGURE 9 – Téléchargement du code source (Bitbucket).

Ce mode de récupération est très simple mais présente un gros inconvénient : il sera impossible de mettre Galaxy à jour de cette façon.

Mercurial

Mercurial est un outil de gestion de versions développé depuis 2005 sous licence GNU GPL. Principalement écrit en Python et utilisable en ligne de commande, Mercurial n'en est pas moins rapide, robuste et facile d'utilisation.

Toutes les commandes de Mercurial commencent par *hg*, formule chimique du mercure.

2. http ://mercurial.selenic.com/
3. https ://bitbucket.org/
4. http ://git-scm.com/

La copie du répertoire distant tient en quelques commandes :

```
mkdir Galaxy
cd Galaxy
hg clone https://bitbucket.org/galaxy/galaxy-dist/
```

Le clonage prend un certain temps. Mercurial doit en effet analyser les nouveautés du répertoire distant et télécharger les fichiers un par un. Le dossier final contient ∼8000 fichiers pour une taille totale d'environ 700 Mo.

L'intérêt d'utiliser Mercurial pour récupérer le code source est de pouvoir mettre ce code à jour, contrairement à Bitbucket.
La mise à jour utilise le système de modification distance de Mercurial. Elle tient donc en deux commandes [5] :

```
hg incoming
hg pull -u
```

La première commande indique si des modifications ont été apportées. Si elle renvoie *no changes found*, la version locale est à jour. Dans le cas contraire, elle renvoie une liste de *changeset* (Fig. 10). La seconde commande peut être exécutée si au moins *changset* est indiqué.

```
changeset:   6528:63bc46cc73b7
tag:         tip
user:        jeremy goecks <jeremy.goecks@emory.edu>
date:        Wed Jan 18 09:48:32 2012 -0500
summary:     Update Tophat tests for v1.4.0
```

FIGURE 10 – Exemple de *changeset* (Mercurial).

2.2.2 Exécution du serveur local

Galaxy fonctionne sur la base d'un serveur local. Cela consiste à ouvrir un serveur fictif sur l'ordinateur de l'utilisateur et à se servir de l'interface proposée par un navigateur web pour accéder aux services proposés par le serveur.

Ce logiciel étant principalement écrit en Python, il est officiellement compatible avec les versions 2.5, 2.6 et 2.7 de l'interpréteur Python par défaut. La version par défaut est donnée en en-tête de l'interpréteur (Fig. 11).

```
Python 2.7.1 (r271:86832, Jul 31 2011, 19:30:53)
[GCC 4.2.1 (Based on Apple Inc. build 5658) (LLVM build 2335.15.00)] on darwin
Type "help", "copyright", "credits" or "license" for more information.
```

FIGURE 11 – Version de l'interpréteur Python.

Pour lancer le serveur local de Galaxy, il suffit d'exécuter le script *run.sh* :

```
sh run.sh # commande de base
sudo run.sh # commande administrateur
```

5. A exécuter dans le répertoire local de Galaxy

La commande "de base" suffit à démarrer le serveur. Il est toutefois conseillé d'utiliser la commande administrateur car certaines opérations [6] nécessitent des droits d'administrateur.

Le serveur charge tous les fichiers requis, configure ses variables et démarre sur une adresse indiquée sur la dernière ligne du terminal (Fig. 12).

```
serving on http://127.0.0.1:8080
```

FIGURE 12 – Adresse du serveur de Galaxy.

Pour stopper le serveur, il suffit d'utiliser *ctrl+z* dans le terminal qui le gère.

Il est possible que le port utilisé par Galaxy soit déjà en utilisation. Dans ce cas, le démarrage du serveur échoue et renvoie une erreur *adress already in use* (Fig. 13). Seul un redémarrage de l'ordinateur peut réinitialiser le port.

```
socket.error: [Errno 48] Address already in use
```

FIGURE 13 – Erreur d'adressage de port.

Une fois le serveur démarré, son accès est possible depuis un navigateur web quelconque (Fig. 14). L'adresse URL à indiquer est l'adresse donnée sur la dernière ligne du terminal.

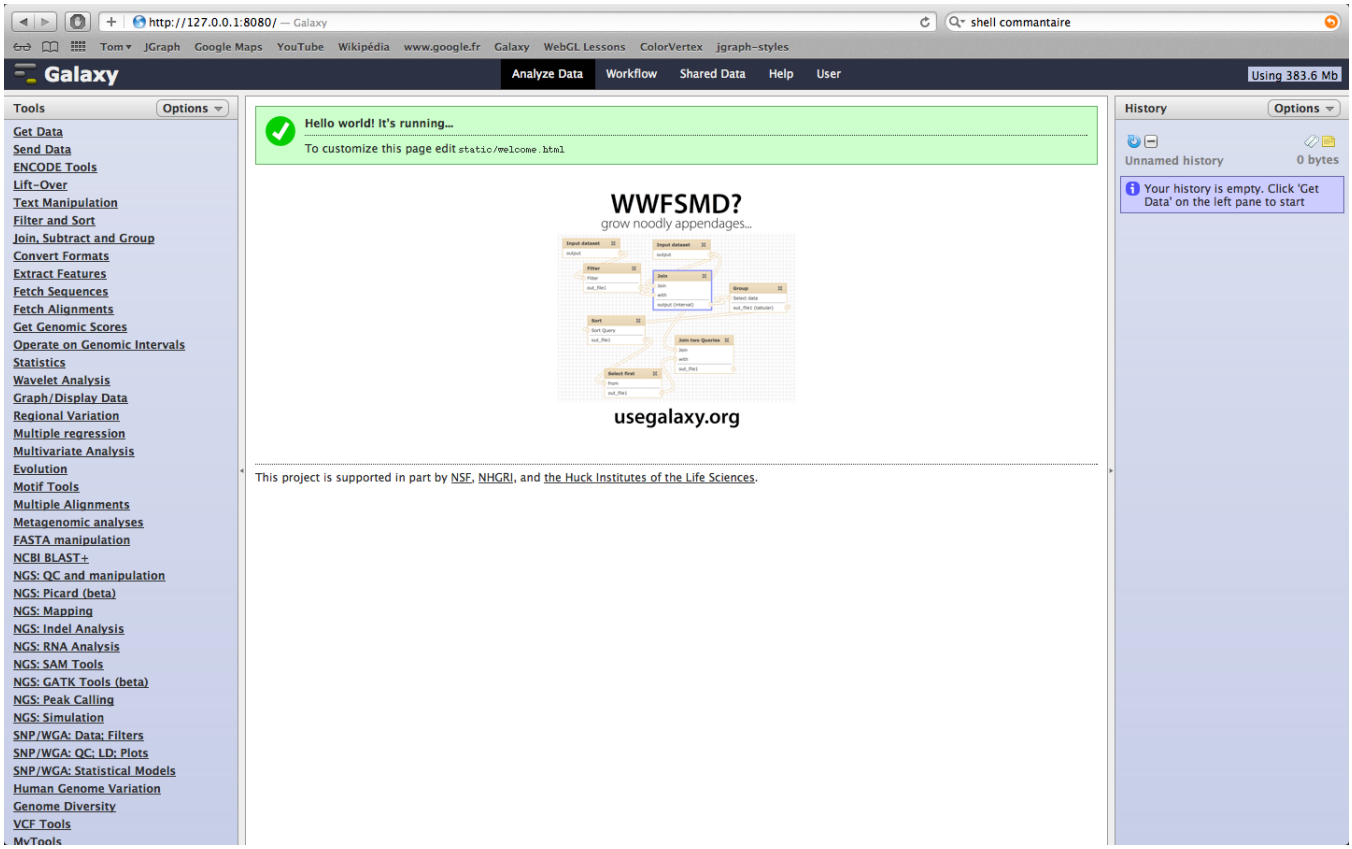

FIGURE 14 – Galaxy en usage local.

6. Notamment l'écriture sur disque dur.

3 Fonctionnalités

Il nous est impossible de recenser et décrire toutes les fonctionnalités disponibles sous Galaxy. Nous nous limiterons donc la description des fonctionnalités aux principales ainsi qu'à l'ajout de plug-ins, permettant ainsi une personnalisation, ainsi que l'utilisation des workflows.

3.1 Présentation générale

Les outils de Galaxy peuvent être classés en quatre grande catégories.

1. Manipulation de fichiers :
   - ouverture de fichiers volumineux,
   - ajout/suppression de lignes,
   - concaténation, filtrage, intersection,
   - etc ...
2. Opérations sur les données :
   - addition, soustraction, moyenne, calcul de taille de séquences,
   - conversion, formatage,
   - etc ...
3. Analyse de séquences :
   - calcul de corrélation,
   - recherche d'orthologues,
   - utilisation des outils d'EMBOSS [1],
   - etc ...
4. Visualisation des données :
   - alignements multiples,
   - distribution de données (histogramme, scatterplot),
   - arbres phylogéniques,
   - etc ...

1. European Molecular Biology Open Software Suite : suite logicielle dédiée aux analyses bioinformatiques

3.2 Ajout de plug-ins

L'ajout de plug-ins est une fonctionnalité uniquement disponible pour Galaxy installé localement. Pour implémenter une fonctionnalité, il faut commencer par développer le script/programme qui fera office de plug-in puis l'intégrer dans la liste des outils.

3.2.1 Création

Même si le cœur de Galaxy est dépendant Python, son architecture lui permet de gérer des scripts écrits dans différents langages de programmation interprétés [2] mais aussi compilés [3]. Cet avantage permet à (presque) tous les programmeurs de développer des outils dans leurs langages favoris, et de les intégrer en toute transparence dans Galaxy.

La communication Galaxy/fichiers est simple car fait intervenir les flux d'entrée/sortie en les redirigeant automatiquement vers :
- les fichiers intégrés dans Galaxy pour le flux d'entrée,
- la fenêtre de résultats pour le flux de sortie.

Il n'est donc pas nécessaire de préparer une configuration spéciale puisque elle est réalisée par Galaxy.

3.2.2 Intégration

Afin de distinguer et de recenser tous les outils qui lui sont disponible, le logiciel possède une base de données de plug-ins sous la forme d'un fichier XML : *tool_ conf.xml*.

tool_ conf.xml agit en réalité comme une liste de pointeur : chaque plug-ins est identifié par un nom, un identifiant et un fichier XML qui correspond à un fichier de configuration. Pour faire savoir qu'un nouveau script est disponible, il suffit d'ajouter les lignes génériques :

```
1 <section name="MyTools" id="mTools">
2     <tool file="myTools/toolExample.xml" />
3 </section>
```

Avec les attributs :
- name : nom de l'outil tel qu'il apparaitra dans l'interface de Galaxy,
- id : identifiant de l'outil,
- file : chemin vers le fichier de configuration XML.

Il est conseillé, pour des raisons pratiques, de créer un répertoire contenant les scripts de l'utilisateur afin de les différencier des autres.
Dans l'exemple précédent, le futur plug-in ainsi que son fichier de configuration (*toolExample.xml*) devront être placés dans un dossier *tools/myTools* que l'utilisateur devra préalablement créer.

2. Tels que : Python, Ruby, Perl, Bash, etc ...
3. Tels que : C, C++, Java, etc ...

toolExample.xml doit au moins contenir les informations suivantes [4] :

```
<tool id="fa_gc_content_1" name="Compute GC content">
  <description>for each sequence in a file</description>
  <command interpreter="perl">toolExample.pl $input $output</command>
  <inputs>
    <param format="fasta" name="input" type="data" label="Source file"/>
  </inputs>
  <outputs>
    <data format="tabular" name="output" />
  </outputs>

  <tests>
    <test>
      <param name="input" value="fa_gc_content_input.fa"/>
      <output name="out_file1" file="fa_gc_content_output.txt"/>
    </test>
  </tests>

  <help>
This tool computes GC content from a FASTA file.
  </help>
</tool>
```

Dans cet exemple, l'outil implémenté est un script Perl qui calcule le GC%. Ce script prend 2 argument lors de son exécution : $input (associé à un fichier interne au format fasta) et $output (associé à la sortie standard de Galaxy) (Fig. 15, 16 et 17).

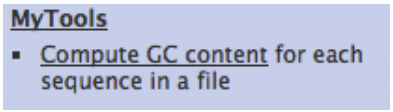

FIGURE 15 – Onglet de l'outil implémenté.

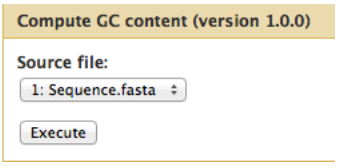

FIGURE 16 – Fenêtre de l'outil implémenté.

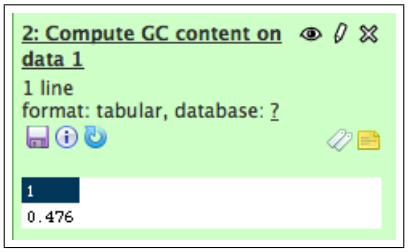

FIGURE 17 – Résultat du script *toolExample.pl*.

Nous pouvons ainsi résumer les opérations nécessaires à l'intégration d'un plug-in de la façon suivante :

1. écrire le script,
2. modifier *tool_conf.xml*,
3. créer le fichier de configuration XML.

4. Basé sur le tutorial : http ://wiki.g2.bx.psu.edu/Admin/Tools/Add%20Tool%20Tutorial

3.3 Workflow

Le workflow Galaxy fournit un ensemble d'outils pour la manipulation et l'analyse de données génomiques. Il est très intuitif dans l'utilisation ce qui en fait une cible de choix pour le biologiste.

Il permet de créer des workflows, les enregistrer dans un espace dédié, les partager, et les exécuter de façon automatique. Les outils dédiés analyse de données NGS sont régulièrement mis à jour.

Galaxy offre donc la possibilité d'exécuter des analyses bioinformatiques sans effort de programmation. La version en ligne est intéressante car elle permet de se familiariser aux logiciels et d'exécuter l'analyse depuis un portable, mais la possibilité d'intégrer ces propres outils est indéniablement un gros avantage de la version locale.

Si nous devions citer un inconvénient, plutôt d'actualité : l'utilisateur est obligé de charger ses données en mémoire dans Galaxy, le temps de chargement peut être très long si l'on manipule des données issues d'expériences NGS, même sur une instance locale. Il est impensable de charger de telles données sur la version web.

Prenons par exemple un workflow de métagénomique (Fig. 18).

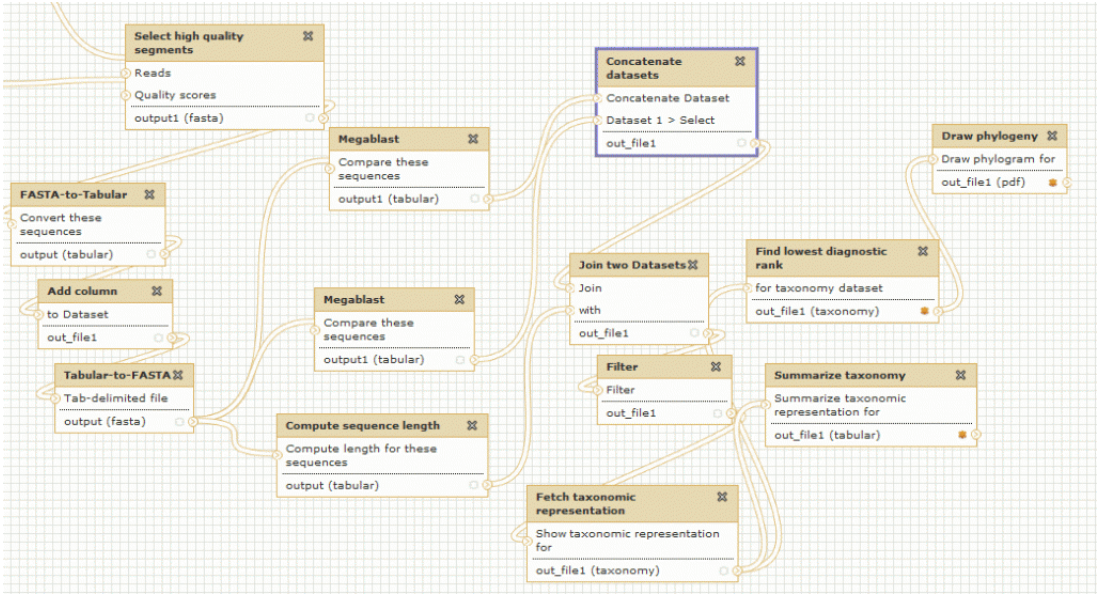

FIGURE 18 – Workflow de métagénomique.

La métagénomique étudie les organismes microbiens directement dans leur environnement sans passer par une étape de culture en laboratoire. Dans ce cas nous cherchons à obtenir des informations relatives à l'origine phylogénétique de l'organisme étudié et aussi des informations taxonomiques.
Pour cela, le workflow prend en entrée des fichiers issu de NGS 454. Dans la première entrée le workflow demande le fichier de read et la deuxième entrée celui de qualité.

Ensuite, ces données passe par un programme chargé de sélectionner les segments de plus grande qualité. Ce programme à comme sortie un fichier au format FASTA. Le principe ensuite est de lancer deux mégaBlast pour comparer les séquences à deux bases de données différentes. Mais pour cela il faut que chaque séquence "de qualité" soit identifiée donc on passe le fichir FASTA sortant du premier programme dans un convertisseur vers un format tabular, on rajoute une colonne qui content le numéro du read et on repasse au format FASTA.

En parallèle on lance un programme qui retourne un tableau avec dans la première colonne le nom de la séquence et dans la deuxième la longueur en acide nucléique. Les sorties des fichiers de blast sont mises dans un même fichier. Le tableau de longueur des séquences est lui aussi agrégé dans un autre programme. Le fichier sortant de ce programme passe dans un nouveau programme qui lui prépare les informations pour l'analyse taxonomique et l'analyse phylogénétique.

Donc la sortie de ce programme est envoyée à deux autres programmes un pour l'analyse phylogénétique et l'autre pour la finalisation de l'analyse taxonomique. Au final on obtient un fichier au format PDF avec l'arbre phylogénétique et un fichier comportant toute les informations taxonomiques.

Ce qui est agréable dans un tel type de workflow est la simplicité de mise en place, surtout avec l'interface graphique ou les programme sont représenté par des boites relié entre elles part des flèches. Ces flèches correspondent aux fichiers qui vont d'une application à un autre.

Conclusion

Les logiciels de Workflow, tels que Galaxy, s'inscrivent dans une logique de pérennisation des processus analytiques, qui a pour but de sortir de la logique « projet » (dans le sens ponctuel) en créant des processus d'analyses génériques. Comme nous l'avons vu, un logiciel de Workflow est un outil permettant d'exécuter un ensemble de processus de façon automatique. Ces « pipelines » sont très présents en bioinformatique (à défaut d'être très utilisés) car ils permettent aux chercheurs en biologie d'analyser leurs données (issues de séquencages, génotypages) de façon relativement transparente et (quasiment) sans l'aide d'informaticiens (denrées rares dans la recherche).

Toutefois, il conviendra de distinguer deux sortes de logiciel de Workflow.

- Les logiciels de Workflow qui permettent aux chercheurs de manipuler leurs données et exécuter leurs analyses sans posséder de connaissances en écriture de scripts ou en bases de données. Les données sont rapatriées au sein du logiciel de Workflow, permettant l'exécution d'un ensemble de tâches, à travers des modules pré-installés. En séquençage, le Workflow permet de convertir des séquences en formats divers, les filtrer ou les assembler. Le logiciel de Workflow ISYS (2001), BioMOBY, Taverna et Galaxy entrent dans cette catégorie.
- Les logiciels de Workflow qui assurent un accès direct à des composants (installés sur le serveur) et/ou aux données génomiques sans passer par un rapatriement préalable des données. WildFire, Pegasys ou Ergatis (ce dernier sera décrit dans un prochain post) font partie de cette catégorie. De manière générale ces logiciels de Workflow sont plus difficiles à prendre en main mais sont évidemment plus flexibles.

Pour résumer, Galaxy permet :

- d'automatiser des processus d'analyse (idéalement répétitifs) en les reliant dans un pipeline,
- de lancer des analyses sur des architectures matérielles complexes telles des grilles de calculs ou des serveurs,
- de formaliser le processus d'analyse en vue d'une publication scientifique.

Bibliographie

[1] E. Afgan, D. Baker, N. Coraor, B. Chapman, A. Nekrutenko, and J. Taylor. Galaxy cloudman : delivering cloud compute clusters. *BMC bioinformatics*, 11(Suppl 12) :S4, 2010.

[2] E. Afgan, D. Baker, N. Coraor, H. Goto, I.M. Paul, K.D. Makova, A. Nekrutenko, and J. Taylor. Harnessing cloud computing with galaxy cloud. *Nature biotechnology*, 29(11) :972–974, 2011.

[3] E. Afgan, D. Baker, A. Nekrutenko, J. Taylor, et al. A reference model for deploying applications in virtualized environments. *Concurrency and Computation : Practice and Experience*.

[4] E. Afgan, J. Goecks, D. Baker, N. Coraor, A. Nekrutenko, and J. Taylor. Galaxy : A gateway to tools in e-science. *Guide to e-Science*, pages 145–177, 2011.

[5] Arveiler. *Génetique humaine*. PhD thesis, Université Bordeaux2, 2011.

[6] D. Blankenberg, N. Coraor, G. Von Kuster, J. Taylor, and A. Nekrutenko. Integrating diverse databases into an unified analysis framework : a galaxy approach. *Database : the journal of biological databases and curation*, 2011, 2011.

[7] D. Blankenberg, A. Gordon, G. Von Kuster, N. Coraor, J. Taylor, A. Nekrutenko, et al. Manipulation of fastq data with galaxy. *Bioinformatics*, 26(14) :1783–1785, 2010.

[8] D. Blankenberg, J. Taylor, A. Nekrutenko, et al. Making whole genome multiple alignments usable for biologists. *Bioinformatics*, 27(17) :2426–2428, 2011.

[9] D. Blankenberg, J. Taylor, I. Schenck, J. He, Y. Zhang, M. Ghent, N. Veeraraghavan, I. Albert, W. Miller, K.D. Makova, et al. A framework for collaborative analysis of encode data : making large-scale analyses biologist-friendly. *Genome research*, 17(6) :960–964, 2007.

[10] C. Bock, G. Von Kuster, K. Halachev, J. Taylor, A. Nekrutenko, and T. Lengauer. Web-based analysis of (epi-) genome data using epigraph and galaxy. *Methods Mol. Biol*, 628 :275–296, 2010.

[11] Laure Buhry. Estimation de paramètres de modèles de neurones biologiques sur une plate-forme de snn (spiking neural network) implantés "in silico". 2010.

[12] EL FAHIME Elmostafa and Ennaji Mly Mustapha. Évolution des techniques de séquençage. *LES TECHNIQUES DE LABORATOIRE N5*, 2007.

[13] B. Giardine, C. Riemer, R.C. Hardison, R. Burhans, L. Elnitski, P. Shah, Y. Zhang, D. Blankenberg, I. Albert, J. Taylor, et al. Galaxy : a platform for interactive large-scale genome analysis. *Genome research*, 15(10) :1451–1455, 2005.

[14] Anat Gluzman and Eran Mick. Algorithms for next generation sequencing. 2011.

[15] J. Goecks, K. Li, D. Clements, J. Taylor, et al. The galaxy track browser : Transforming the genome browser from visualization tool to analysis tool. In *Biological Data Visualization (BioVis), 2011 IEEE Symposium on*, pages 39–46. IEEE, 2011.

[16] R. Lazarus, J. Taylor, W. Qiu, and A. Nekrutenko. Toward the commoditization of translational genomic research : Design and implementation features of the galaxy genomic workbench. *Summit on translational bioinformatics*, 2008 :56, 2008.

[17] W. Miller, K. Rosenbloom, R.C. Hardison, M. Hou, J. Taylor, B. Raney, R. Burhans, D.C. King, R. Baertsch, D. Blankenberg, et al. 28-way vertebrate alignment and conservation track in the ucsc genome browser. *Genome research*, 17(12) :1797–1808, 2007.

[18] S.K. Pond, S. Wadhawan, F. Chiaromonte, G. Ananda, W.Y. Chung, J. Taylor, A. Nekrutenko, et al. Windshield splatter analysis with the galaxy metagenomic pipeline. *Genome research*, 19(11) :2144–2153, 2009.

[19] M.C. Schatz. The missing graphical user interface for genomics. *Genome biology*, 11(8) :128, 2010.

[20] J. Taylor, I. Schenck, D. Blankenberg, and A. Nekrutenko. Using galaxy to perform large-scale interactive data analyses. *Curr Protoc Bioinformatics*, 10(10.5), 2007.

[21] Tom Walsh and Hashem Shahin. Whole exome sequencing and homozygosity mapping identify mutation in the cell polarity protein gpsm2 as the cause of nonsyndromic hearing loss dfnb82. *The American Journal of Human Genetics*, 2010.