Master Bioinformatique – Université de Rouen Normandie
M1 Semestre 2 – mis à jour janvier 2020
UE3 Bioinformatique en sciences omiques 1 – matière **web services et annotations**
H. Dauchel – A. Lefebvre. et T. Lecroq

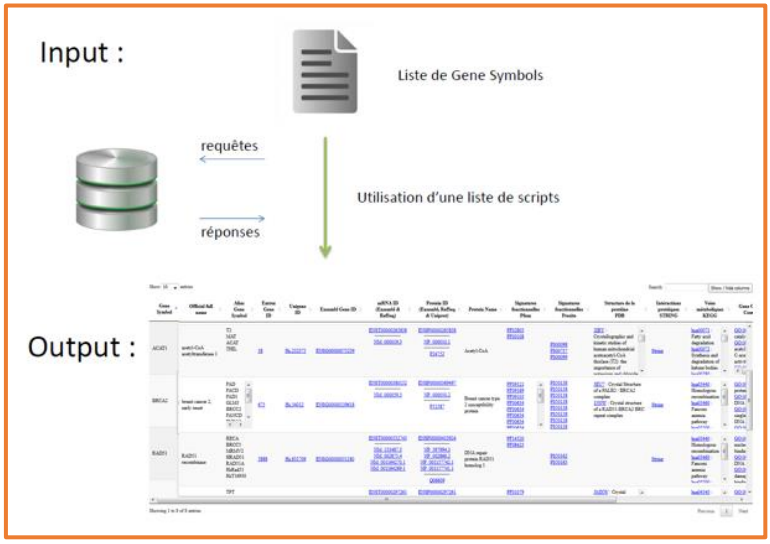
Projet « scripting pour l'agrégation automatique d'annotations »

Modalité

Le travail est individuel* et personnel mais les échanges ne sont pas interdits.
Le travail final fera l'objet d'une présentation- démonstration individuelle (Livraison).
Le travail est réalisé en autonomie avec des points bilans d'avancement (et une interaction technique avec les enseignants informaticiens).
*Le travail d'interfaçage sera un plus si le travail de script est fini, il se réalisera obligatoirement en binome ou trinome.

Cahier des charges

1. Vous mettrez en place une série de scripts permettant d'agréger à partir d'une liste de gènes d'une espèce donnée, leurs annotations respectives dans un fichier tabulé interactif.
2. Optionnel (si 1 fini) : développement d'une interface, en équipe binome ou trinome



Quelles annotations ?

En bleu : données de départ (input)
En rouge : sources primaires
En noir, liste des informations requises dans le tableau final (output)

Informations générales : ADN-ARN(s)-Protéine(s)

- ✓ **Gene Symbol** (ex : RAD51 sur **Gene, NCBI**) pour l'organisme ***Genre species*** (ex : *Homo sapiens*)
- ✓ Offical full name (Ex : RAD51 recombinase, Gene, NCBI)
- ✓ Gene access number : (ex : 5888 sur Gene, NCBI ; sur Ensembl: (ex : ENSG00000051180) + le lien de visualisation sur le genome browser Ensembl et UCSC
- ✓ RNA access number(s)(ex : NM_001164269.1. sur RefSeq, ENST00000267868 sur Ensembl)
- ✓ Protein name(s) (ex : DNA repair-protein RAD51 homolog 1 sur UniprotKB)
- ✓ Protein acces number(s) (ex : Q06609 sur UniprotKB, NP_001157741.1. sur RefSeq, ENSP0000026786 sur Ensembl)

Annotation fonctionnelle et structurale de la protéine

- ✓ Functional signature(s) (ex : PF08423 domaines protéiques sur Pfam + graphical view, ex : PS50162 motifs et domaines sur Prosite + graphical view)
- ✓ 3D Protein Structure(s) : (ex : 1b22, N-terminal Domain sur PDB ou autre banque de structures)

Annotation relationnelle :

- ✓ Gene ontologies (ex : Function : transcription; Biological process : DNA damage ; Cellular component : nucleus sur GO)
- ✓ Pathways (ex : hsa:5888 + les voies hsa03440 Homologous recombination sur KEGG ou Reactome)
- ✓ Protein Interactions (ex : lien vers String ou IntAct)
- ✓ Orthologous gene(s) (Ensembl Compara)

Contraintes de la solution

- ✓ **Le modèle de la collecte des annotations est fourni (figure 1). Vous aurez à le précisez avec le nom de vos méthodes et scripts.** L'origine de l'annotation est dès que possible la source primaire de l'information.
- ✓ Pour agréger les annotations, vous utiliserez les connaissances acquises au cours de vos enseignements en réinvestissant obligatoirement une **diversité d'outils de scripting et web services** : programmation Python et bioPython; API (API REST Ensembl, API e-utilities du NCBI), requêtes SQL ; outils de *ID mapping ;* HTML et construction d'URL, etc.
- ✓ Votre solution devra fonctionner **quelle que soit l'espèce (voir exemples des fichiers input fournis)**
- ✓ Le tableau interactif sera construit avec le **plug-in DataTables de la librairie jQuery en Javascript**: il comportera donc les liens .html fonctionnels vers les banques ressources et les autres facilités d'utilisation interactive (fonction de tri, recherche, ascenseur, personnalisation de l'ordre et de l'affichage des colonnes…). **(voir exemples des fichiers output fourni)**

Environnement et phase de travail

- ✓ Phase d'analyse :
  - repérage des liens croisés entre les portails et banques
  - recherche de documentations sur l'accès programmatique aux banques de données (web services)
  - repérage du fonctionnement du plug-in DataTables
- ✓ Phase de développement : Mise en œuvre de la programmation des scripts
  - Conseils : (figure 2)
    - Organisation **modulaire** selon les bases de données
    - Lancement par un script principal
    - Structure de données pour la collecte des annotations
- ✓ Phase d'interfaçage : travail en binome ou trinome, un seul style d'interface, chacun connectant ses scripts dans un onglet.

Livraisons

- ➢ **Livrable : Semaine 10 (date à préciser) –** espace dépôt Moodle

Un fichier de la forme Prenom_Nom_Annotation.tar.gz comprenant :

- Le fichier final de votre modélisation de la forme Prenom_Nom_schema_conceptuel.pdf
- Le diaporama de votre présentation (Cf ci-dessous) Prenom_Nom_Presentation.pdf
- Les sources de vos scripts de la forme script.py (ou autre si autre langage)
- Un fichier input comportant un exemple d'un petit jeu de données (10 Gene symbols) de la forme Genesymbols.txt
- Un fichier output correspondant de la forme Results.html
- (Option) L'application interfacée déployable

- ➢ **Présentation individuelle : Semaine 10 (créneaux à préciser) ; 10 min + 5 min questions**

Pas d'introduction !

1. Votre solution (livrable) :
   - **D1 : Modélisation** > schéma général de votre collecte d'informations d'annotations avec les méthodes
   - **D2 : Organisation du livrable**, ie du fichier source (fichier input, organisation des modules, script principal, fichier sortie) ;
   - **D3 : lancement de la démonstration en ligne de code et sur l'interface (si réalisée)**
2. **[D4-D8 max] Présentation de portions de votre code**: un exemple pour chaque type de solution de script (API, e-utilities construction d'URL, requête SQL etc.) avec arguments et paramètres ; structure de données collectées et création du tableau de sortie
3. **(Option) D9 Présentation technique de l'interface**
4. **D10 (max) Conclusions/ Autoréflexions**: difficultés et contournements, points positifs/négatifs (complet ou pas, temps d'exécution…).