Etude des éléments transposables du génome Arabidopsis Thaliana.

Akshey MOHAMMED KHOKAN[1], Hugo FERNANDES TELES[1], Maxime COUSTOU[2]
[1]Etudiants en Licence Double-Diplôme : Biologie et Informatique
[2]Etudiant en Licence : Génomique, Biologie et Informatique

Introduction

Les éléments transposables (ET) sont des séquences répétées dans le génome pouvant se multiplier et se déplacer au sein du génome. Il existe plusieurs types d'ET, notamment les transposons à ADN qui fonctionnent sur le principe de couper-coller ainsi que les rétrotransposons avec ou sans LTR qui fonctionnent tous deux sur le modèle copier-coller. On étudie dans ce projet les éléments transposables de l'arabette (Arabidopsis thaliana), un organisme modèle dont le génome est étudié en détail par les chercheurs, étant donné qu'il s'agit du premier génome de plante séquencé. Cette espèce a un génome relativement petit (119 millions de pb d'après le modèle TAIR10).

L'objectif de ce projet est de concevoir une base de données regroupant l'ensemble des copies, ainsi que les copies de référence, de six familles d'éléments transposables issues du génome TAIR10 de l'Arabidopsis thaliana, sélectionnées par nos soins. Cette base intégrera des informations issues de l'analyse BLAST, permettant de caractériser chaque copie, ainsi que leur annotation. Elle sera enrichie par des données sur les gènes et les chromosomes associés.

Grâce à l'annotation des copies d'ET, on cherche à étudier les variabilités du génome ainsi que le fonctionnement de ces éléments notamment dans leurs mécanismes de réplication et d'éventuelle régulation de l'expression des gènes proches.
À travers cette étude, on cherche donc à étudier les copies d'éléments transposables en essayant de savoir si elles sont présentes dans des régions où il y a des gènes, si ces copies sont complètes ou s'il y a eu dégradation par rapport à la séquence de référence. On pourra alors déterminer si certaines copies sont fonctionnelles ou non.

La création de cette base de données permet de répondre aux différentes questions biologiques évoquées précédemment. Elle pourrait être un outil précieux pour les chercheurs en génétique et biologie moléculaire souhaitant analyser l'impact des transposons sur l'expression des gènes. Les bio-informaticiens pourraient également s'en servir pour intégrer d'autres types de données de manière complémentaire. Enfin, des spécialistes en génomique pourraient l'exploiter pour étudier l'évolution des éléments transposables et leur influence sur les génomes.

1. Matériel et méthodes.

1.1 Matériel

Pour débuter notre projet, nous avons récupéré les séquences de toutes les copies d'ET des familles que nous avons sélectionnées grâce au fichier 'TAIR10_TE.fas'. Le fichier 'Athaliana_RepBase_EMBL.txt' nous a donné accès aux séquences consensus de chaque famille, utilisée plus tard comme comparaison avec les séquences des copies. Par ailleurs, nous avons également utilisé le fichier 'TAIR10_Transposable_Elements.txt' pour avoir des informations supplémentaires comme l'orientation, position dans le génome, famille et super-famille de nos copies. Le fichier TAIR10_genes_positions.txt nous apporte par ailleurs les informations sur le génome. Nous avons récupéré ces fichiers par e-campus, bien qu'il aurait été possible de trouver ces séquences en ligne.

1.2 Méthodes/Outils

Nous avons utilisé des commandes Unix pour sélectionner nos familles, prendre toutes les copies d'une famille dans le fichier des séquences fasta et connaître certaines informations utiles telles que :

- le nombre de copies d'ET par famille:

```
awk 'NR > 1 {print $5}' TAIR10_Transposable_Elements.txt | sort | uniq -c
```

- le nombre total de familles:

```
awk 'NR > 1 {print $5}' TAIR10_Transposable_Elements.txt | sort | uniq | wc -l
```

- la liste de toutes les familles:

```
awk 'NR > 1 {print $5,$6}' TAIR10_Transposable_Elements.txt | sort | uniq
```

Notre étude se porte sur 6 familles d'ET :

- ATLINE1_3A - rétrotransposon Non-LTR (LINE)
- ATLINEIII - rétrotransposon Non-LTR (LINE)
- ENDOVIR1 - rétrotransposon LTR (Copia)
- META1 - rétrotransposon LTR (Copia)
- SIMPLEGUY1 - transposon ADN (Harbringer)
- VANDAL11 - transposon ADN (MuDR)

Pour comparer les copies à leur référence, nous avons d'abord transformé les formats EMBL de nos familles (trouvées avec CTRL+F) dans Athaliana_RepBase_EMBL.txt en formats fasta grâce à un programme Python "embl-to-fasta.py". Nous avons ensuite utilisé BLAST+ en local avec des commandes Unix pour comparer les copies à leur séquence de référence grâce aux fichiers fasta. Un script python a également été utilisé pour afficher les alignements résultants du blastn. Ces familles ont été sélectionnées car elles contiennent toutes au moins 15 copies différentes. A partir du fichier Athaliana RepBase EMBL, on peut retrouver les séquences ainsi que 2 annotations de nos 6 séquences de référence. Cependant, certaines de ces séquences ont été annotée il y a plusieurs années et par conséquent leurs annotations sont plutôt faibles.

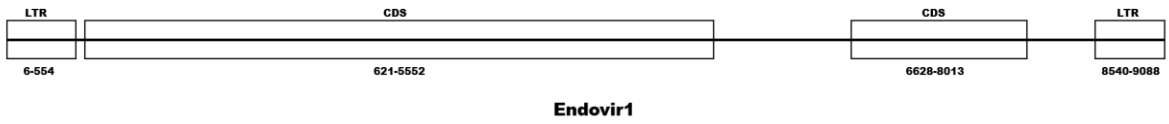


Figure 1A: Structure de la copie de référence Endovir1

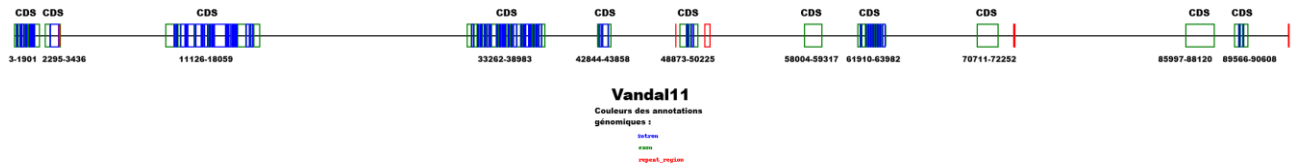
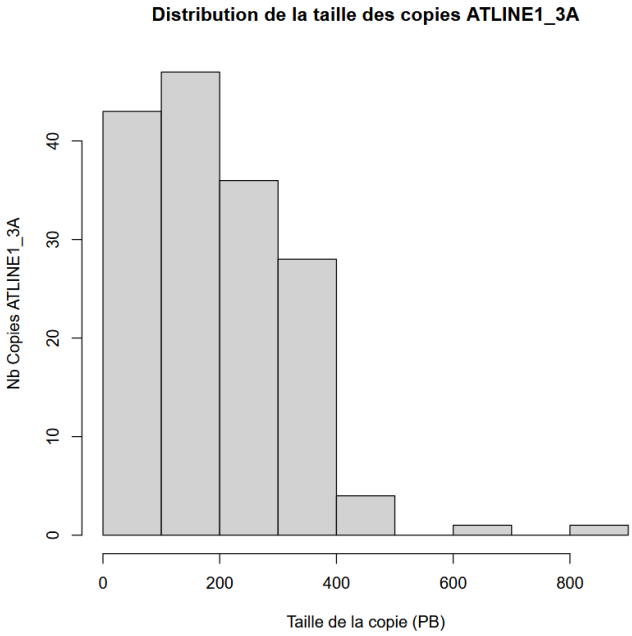


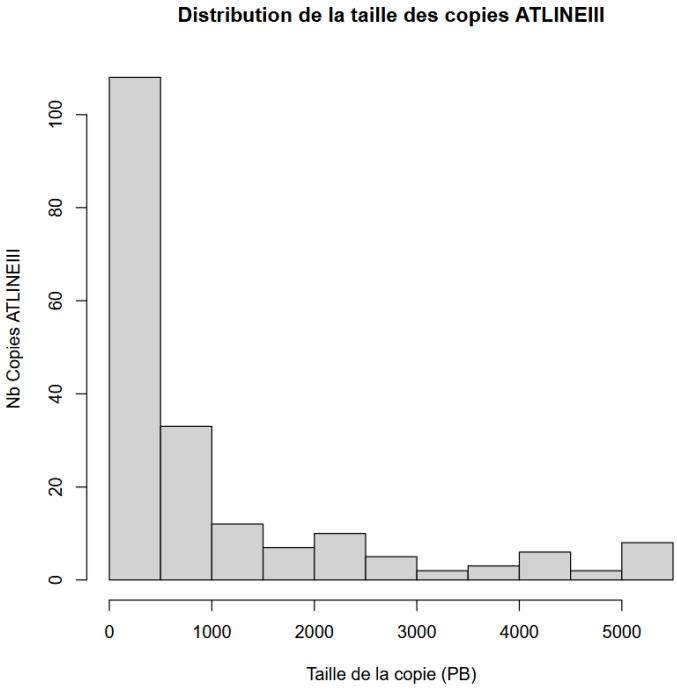
Figure 1B : Structure de la copie de référence Vandal11

2. Résultats et Analyses Blast.



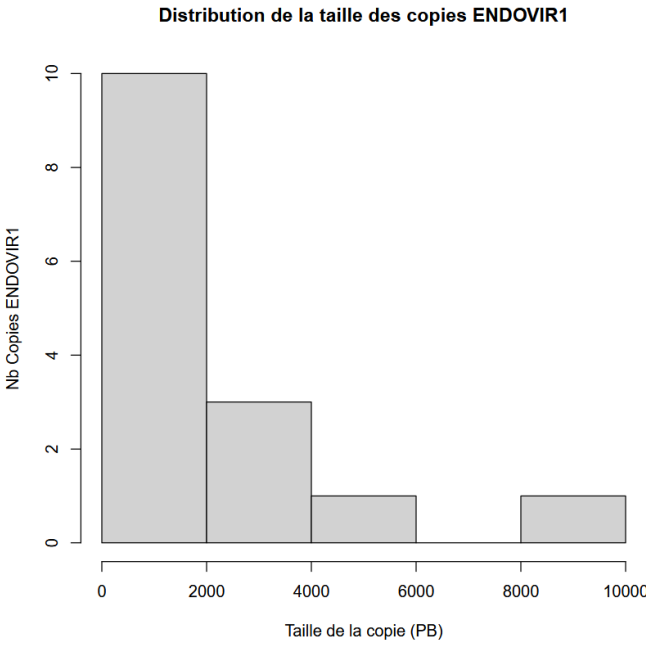
La copie de référence pour ATLINE1_3A fait 330 pb, il y a donc un majorité des copies qui sont plus courtes (<200pb). On remarque qu'il y a environ 26 copies qui correspondent à la taille attendue et qui sont donc potentiellement fonctionnelles.

ATLINE1_3A : 160 copies au total



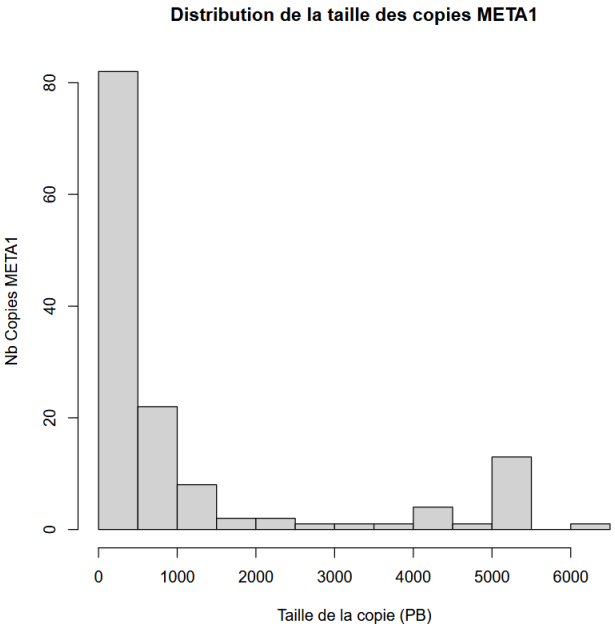
La copie de référence pour ATLINEIII fait 5408 pb, il y a donc très peu de copies de la taille attendue. La plupart sont délétées mais environ 10 copies sont susceptibles d'être fonctionnelles.

ATLINEIII : 196 copies au total



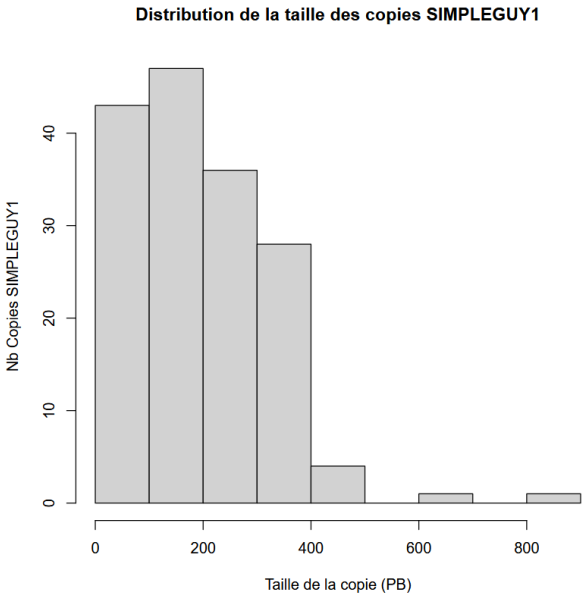
La copie de référence pour ENDOVIR1 fait 9083 pb. Seule une copie est de la taille attendue et pourrait donc être fonctionnelle.

ENDOVIR1 : 15 copies au total



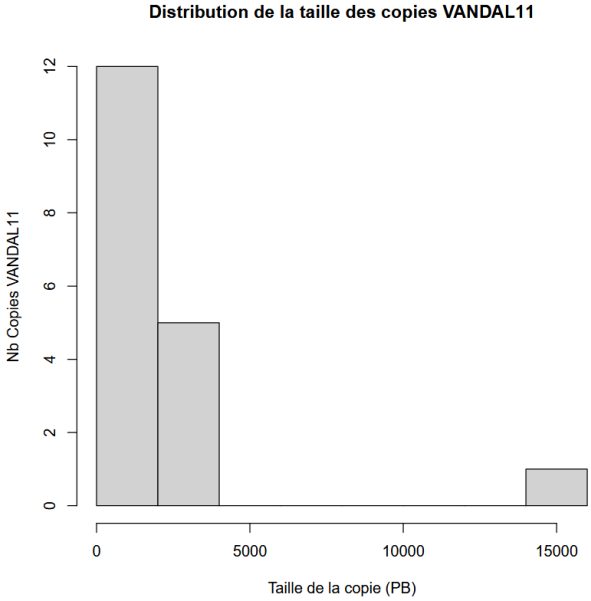
La copie de référence pour la famille META1 fait 5145 pb. La majorité des copies sont délétées mais une quinzaine d'entre elles font la taille attendue et sont potentiellement fonctionnelles.

META1 : 138 copies au total



La copie de référence de SIMPLEGUY1 est de 1068 pb. On remarque que dans cet histogramme, aucune copie ne semble être complète.

SIMPLEGUY1 : 116 copies au total



La copie de référence de VANDAL11 fait 14 166pb. Il y a une copie potentiellement fonctionnelle.

VANDAL11 : 18 copies au total

Suite à l'exécution de BLAST en ligne de commande, nous avons obtenu les meilleurs alignements pour chaque famille d'éléments transposables. Les trois alignements les plus significatifs sont présentés en annexe. Cette analyse nous a permis de comparer les séquences de référence de nos 6 familles de transposons avec toutes les copies identifiées. Certaines copies présentent des divergences notables par rapport à leur séquence de

référence, en raison d'événements évolutifs tels que des mutations ou duplications/délétions de segments. On parle de copie fonctionnelle lorsqu'elle conserve sa capacité à se déplacer et à se répliquer dans le génome d'Arabidopsis thaliana. Généralement, une copie très proche de sa référence en termes de taille et d'identité de séquence est considérée comme fonctionnelle. En annexe, nous avons mis 1 copie/famille car nous avons choisi 6 familles.

3. Base de données relationnelle.

3.1 Spécifications

Une base de données est un système organisé permettant de stocker, structurer et gérer efficacement de grandes quantités d'informations. Elle permet de retrouver rapidement des données précises à l'aide de requêtes. Les BDD sont généralement administrées par des systèmes de gestion de base de données comme MySQL, PostgreSQL ou MongoDB. Cependant, dans le cadre de notre projet, nous avons utilisé PostgreSQL simultanément avec PgAdmin4 afin d'avoir une interface graphique de notre BDD.

Pour uniformiser nos résultats, éviter la redondance des données, offrir un espace de recherche de données plus intuitif et plus organisé à grande échelle, nous souhaitons stocker les informations sur les copies d'éléments transposables de l'Arabidopsis thaliana. En effet, les copies ont été enregistrées avec leur numéro d'accession, leur taille et la séquence consensus à laquelle ils font référence, et si elles sont fonctionnelles ou non. De plus, les familles auxquelles elles appartiennent ont été enregistrées (avec le nom, leurs nombres d'éléments transposables, et la classe à laquelle elles appartiennent). Cela nous permettra d'étudier plus en profondeur les 6 familles que nous avons choisies. Les chromosomes sur lesquels se trouvent les copies sont aussi enregistrés, avec leur numéro, le génome auquel ils appartiennent et leur taille. Les gènes se trouvant sur ces chromosomes ont été aussi enregistrés, avec leur nom d'accession, leur fonction et leur taille. Pour déterminer si nos copies sont potentiellement fonctionnelles, les résultats

BlastN ont été enregistrés, avec le % d'identité, le nombre de hits, et la taille de l'alignement.

3.2 Modèle Entité Association

Le modèle EA est un outil conceptuel qui sert à modéliser les données d'un système de manière claire et logique, avant leur implémentation dans une base de données relationnelle.

Il repose sur trois éléments principaux :

- Les entités : objets sur lesquels on veut stocker des informations
- Les attributs : caractéristiques de ces objets
- Les associations : liens entre entités

Ce modèle précise aussi les cardinalités.

En effet, dans le cadre de notre projet, nous avons identifié quatre types d'entités principales :

Chromosome

Clé primaire : Numéro

Copie

Clé primaire : NuméroAccession

Famille ET

Clé primaire : Nom

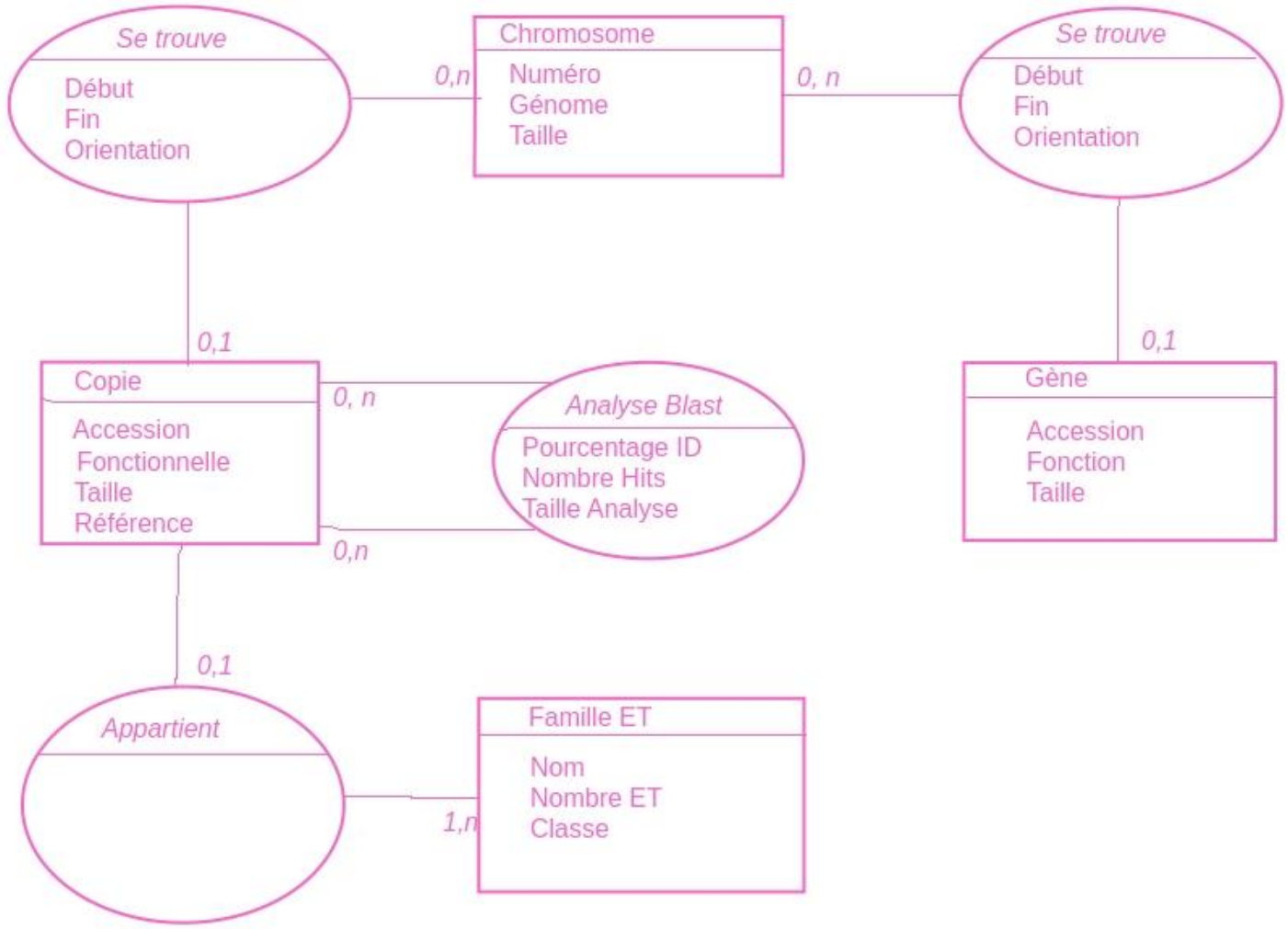
Gène

Clé primaire : NuméroAccession

L'association "Appartient" relie les entités Copie et Famille ET selon une relation de type one-to-many. Plus précisément, la cardinalité du côté de Copie est de 0,1, ce qui signifie qu'une copie peut appartenir à zéro ou une seule famille d'éléments transposables. En revanche, la cardinalité du côté de Famille ET est de 0,n, indiquant qu'une même famille peut être associée à plusieurs copies. Ainsi, cette association traduit une relation d'appartenance non obligatoire mais potentiellement multiple du côté de la famille, ce qui permet de structurer les copies selon une classification hiérarchique.

En ce qui concerne l'association "Se trouve", elle relie les entités Gène et Chromosome dans une relation de type many-to-one. La cardinalité du côté Chromosome est de 0,n, ce qui signifie qu'un chromosome peut contenir plusieurs gènes. De son côté, la cardinalité

0,1 du gène indique qu'un gène peut être localisé sur zéro ou un seul chromosome. Cette association permet donc de modéliser la localisation génomique des gènes, en tenant compte du fait qu'ils peuvent ne pas être toujours positionnés dans le contexte étudié. Enfin, l'association réflexive "Analyse BLAST" s'applique à l'entité Copie elle-même, selon une relation de type many-to-many réflexive. Cette association permet de représenter les comparaisons de similarité entre différentes copies, telles qu'obtenues par BLAST. Chaque occurrence de cette relation nécessite l'utilisation de deux clés étrangères (une pour chaque copie de l'association réflexive), et il est impératif de leur attribuer des noms distincts afin d'éviter toute ambiguïté dans le passage au modèle relationnel logique, où tous les attributs doivent être identifiables de manière unique.



3.3 Schéma logique relationnel

CHROMOSOME (numero [PK], genome, taille)
GENE (accession [PK], fonction, taille)
SE_TROUVE_GENE (accession_gene [PK, FK], numero_chromosome [PK, FK], debut, fin, orientation)
COPIE (accession [PK], fonctionnelle, taille, reference)
SE_TROUVE_COPIE (accession_copie [PK, FK], numero_chromosome [PK, FK], debut, fin, orientation)
FAMILLE_ET (nom [PK], nombre_et, classe)
APPARTIENT (accession_copie [PK, FK], nom_famille [FK])
ANALYSE_BLAST (id [PK], accession_copie [FK], pourcentage_id, nombre_hits, taille_analyse)

Dom(CHROMOSOME.numero) ⊇ Dom(SE_TROUVE_GENE.numero_chromosome)
Dom(GENE.accession) ⊇ Dom(SE_TROUVE_GENE.accession_gene)
Dom(CHROMOSOME.numero) ⊇ Dom(SE_TROUVE_COPIE.numero_chromosome)
Dom(COPIE.accession) ⊇ Dom(SE_TROUVE_COPIE.accession_copie)
Dom(COPIE.accession) ⊇ Dom(APPARTIENT.accession_copie)
Dom(FAMILLE_ET.nom) ⊇ Dom(APPARTIENT.nom_famille)
Dom(COPIE.accession) ⊇ Dom(ANALYSE_BLAST.accession_copie)

3.4 Implémentation de la base de données

Nous utilisons le système de gestion de base de données relationnelle PostgreSQL. Cependant, afin d'avoir une visualisation graphique, l'utilisation de PgAdmin4 a été réalisée. De plus, nous avons eu pour idée d'intégrer toutes les copies, toutes les familles, tous les chromosomes et gènes mis à notre disposition afin d'enrichir notre base de données et de permettre de répondre aux problématiques biologiques.
Afin de réaliser cette démarche, nous avons extrait les informations nécessaires à chaque table de notre BDD et nous les avons intégrées à des fichiers via Linux. Cela a permis une implémentation efficace de toutes les données dans la base.
Les codes ayant permis cette implémentation se trouvent en annexe.

3.5 Bilan du nombre de données implémentées dans la base de données

Nb_Hits	215

Chromosomes	7 (dont génome du chloroplaste, et génome mitochondrial)
Nombre copies	31189
Familles	320
Gènes	27628

Figure 2: Tableau récapitulatif du nombre de données implémentées dans la BDD

Le nombre de hits s'explique par le fait que seuls 6 BlastN ont été faits.

3.6 Requêtes intéressantes de la base de données

SELECT accession, nom_famille, taille FROM appartient JOIN copie AS co ON accession_copie = co.accession WHERE co.fonctionnelle = 't';

	accession character varying (20)	nom_famille character varying (50)	taille integer
1	AT1TE21210	META1	5137
2	AT5TE79680	SIMPLEGUY1	1051
3	AT5TE63610	ENDOVIR1	9088
4	AT4TE21900	VANDAL11	14215
5	AT2TE01000	ATLINEIII	5408
6	AT5TE73020	SIMPLEGUY1	1071
7	AT1TE14315	META1	5129
8	AT1TE21710	META1	5213
9	AT3TE61000	META1	5275
10	AT3TE76010	META1	5150

Figure 3A: Requête SQL affichée via pgAdmin4

On obtient ici les copies fonctionnelles annotées manuellement avec leurs familles respectives et leurs tailles.

SELECT te.accession_copie, tg.accession_gene, gene.fonction, te.numero_chromosome, te.debut, te.fin, te.orientation, tg.debut, tg.fin FROM se_trouve_copie AS te JOIN se_trouve_gene AS tg ON tg.numero_chromosome = te.numero_chromosome JOIN copie ON copie.accession = te.accession_copie JOIN gene ON gene.accession =

tg.accession_gene WHERE tg.orientation = te.orientation AND copie.fonctionnelle = TRUE AND LEAST(tg.fin, te.fin) - GREATEST(tg.debut, te.debut) > -1000 ORDER BY te.accession_copie ASC, tg.numero_chromosome ASC LIMIT 1000;

	accession_copie character varying (20)	accession_gene character varying (20)	fonction text	numero_chromosome character varying (2)	debut integer	fin integer	orientation character varying (2)	debut integer	fin integer
1	AT1TE14315	AT1G12930	description:ARM	1	4405996	4411119	-	4398322	4405669
2	AT1TE21210	AT1G18940	description:Nodulin-like	1	6538419	6543539	+	6543781	6545802
3	AT1TE21710	AT1G19394	[null]	1	6704197	6709404	+	6709773	6710670
4	AT1TE21710	AT1G19396	[null]	1	6704197	6709404	+	6709778	6710345
5	AT2TE01000	AT2G01540	description:Calcium-dependent	2	243804	249211	-	242041	243649
6	AT3TE76010	AT3G50480	description:homolog	3	18734249	18739393	+	18733042	18734388

Figure 3B: Requête SQL affichée via pgAdmin4

On obtient ici les copies fonctionnelles qui sont chevauchantes, en amont ou en aval (1000pb) d'un gène sur le même chromosome et ayant la même orientation, afin de voir si il y a régulation de l'expression génique. On peut voir que deux gènes ici sont sans fonction et pourtant la copie qui est près de ces gènes.

4. Conclusion et discussions.

Au cours de ce projet, nous avons utilisé plusieurs outils bio-informatiques pour explorer les éléments transposables (ET) chez Arabidopsis thaliana, à partir des fichiers fournis en version TAIR10 du génome. Il est important de noter que cette version est ancienne, et que l'utilisation de versions plus récentes pourrait modifier les résultats obtenus : par exemple, la séquence de référence pour VANDAL11 fait 14 166 pb dans notre fichier, alors que la même accession sur GenBank correspond à une séquence de 96 256 pb.
Nous avons d'abord appliqué des commandes Linux pour extraire les informations clés (familles, positions, chromosomes…), puis constitué deux jeux FASTA : un avec les copies d'ET des familles choisies (LTR, non-LTR, ADN), l'autre avec leurs références. Grâce à BLASTN, nous avons comparé les copies aux références pour évaluer leur conservation et fonctionnalité potentielle.
Notre base de données SQL n'a pas été limitée aux seules familles étudiées. Nous avons intégré toutes les copies disponibles dans les fichiers d'origine, en structurant les données autour des familles, des chromosomes, des gènes et des relations entre ces entités. Cette base a été visualisée et interrogée via une interface graphique grâce à pgAdmin4, ce qui facilite considérablement l'exploration des données.

Ce projet a donc été une occasion concrète de mettre en œuvre des compétences en traitement de données biologiques, en modélisation, en programmation et en gestion de base de données, au service d'une problématique réelle de génomique fonctionnelle.

5. Perspectives

Plusieurs perspectives se dégagent à partir de ce travail :
Automatiser l'analyse structurale des ET afin d'enrichir la base de données avec des critères de fonctionnalité prédictive, utilisables dans des requêtes SQL ou via l'interface pgAdmin.
Mettre à jour les jeux de données en utilisant des versions plus récentes du génome d'Arabidopsis thaliana, afin de refléter au mieux les connaissances actuelles et éviter les biais dus à des annotations obsolètes.
Étendre la base de données pour intégrer des informations sur les gènes voisins, permettant d'explorer les liens potentiels entre ET et régulation génique, voire des pathologies associées dans d'autres espèces.

6. Annexe

https://github.com/crakshay1/TEnder