

## PlateForme **TERRIERSIR2017**.

### Etapes à suivre :

- 
- Ouvrir le projet dans un environnement de développement Java, comme Eclipse.
- Dans la classe : `org.terrier.utility.ApplicationSetup`, modifier les paths dans les lignes suivantes au milieu du code, avec le chemin complet du projet comme suit :
 

```
setProperty("terrier.home", "/users/ines/Bureau/WorkspaceValentina/TERRIERSIR2017/");
setProperty("terrier.etc", "/users/ines/Bureau/WorkspaceValentina/TERRIERSIR2017/etc/");
setProperty("terrier.setup", "/users/ines/Bureau/WorkspaceValentina/TERRIERSIR2017/etc/terrier.properties");
propertiesFile = getProperty("terrier.setup", "/users/ines/Bureau/WorkspaceValentina/TERRIERSIR2017/etc/terrier.properties");
```
- Dans le fichier `TERRIERSIR2017/etc/terrier.properties` : changer les paths concernant les requêtes, les qrels, etc, le `weightingModel` (TF\_IDF, BM25...), ainsi que les tags à tenir en compte et à ignorer pour les documents (exemple de tags pour les recettes de cuisine : RECIPE, TI, IN, PR, NUM) et pour les requêtes (exemple : num, title, desc, narr, etc.).
- Dans le fichier `TERRIERSIR2017/Documents/chemin.properties` : modifier les paths (des ontologies) également avec les vôtres, ainsi que les paths des annotations, etc.
- Il y a 1 dossier source ajouté : `NewSrc`, dans lequel il y a 3 package : Annotation, Application et SemanticIR (c'est celui qui contient le main "MainRIS.java").
- Le fichier `"/etc/collection.spec"` est le fichier là où il faut mettre le chemin complet de la collection documentaire. Si c'est un seul fichier (exemple `RecipeBase.xml`), mettre directement le chemin du fichier. S'il s'agit d'une collection dont les documents sont dans plusieurs dossiers imbriqués (le cas de la collection TRECWT10G), alors commencer par exécuter la classe : `"/src/core/org.terrier.applications/FileFind.java"` en donnant en arguments au "main" lors de l'exécution le chemin du dossier contenant tous les dossiers de la collection. Le résultat sera le remplissage du fichier "collection.spec" avec les chemins de tous les fichiers contenus dans ces dossiers (exemple pour TREC WT10G, les chemins des fichiers `B*.gz`). Dans cette classe "FileFind.java" il faut aussi modifier le chemin complet de dans la ligne :
 

```
Ecrire("/users/ines/Bureau/WorkspaceValentina/TERRIERSIR2017/etc/collection.spec", file);
```
- Finalement pour exécuter il faut aller dans `mainRIS.class` et exécuter en donnant en arguments, pour l'indexation "-i" puis "-v" (pour faire les fichiers inversés), pour la recherche "-r", pour l'évaluation "-e".
- Il y a des modifications à faire pour passer d'une RI classique, notamment ces modifications à faire :
  - Dans la classe : `org.terrier.matching.taat.Full.java` (qui est la classe du matching Modèle), il y a une variable booléenne **Sem** à mettre à true ou false selon si on veut une méthode de RI classique ou RIS (sans oublier bien sûr dans le fichier `terrier.properties` de modifier la méthode de pondération `WeightingModel`, exemple à `SemBM25`).
  - Dans la classe `org.terrier.matching.PostingListmanager` : c'est là qu'il faut indiquer l'ontologie utilisée dans le constructeur de la classe.
  - D'autres petites modifications sont à faire pour certaines méthodes : (voir dans les `READ.txt` du dossier des expériences qui sont dans le dossier **ResultatsTerrier2017**).