TP1

Intégration de données avec Talend Open Studio for DI

©Mourad Ouziri
Mourad.Ouziri@u-paris.fr

Programme-objectifs :

- Ingestion de données de différents formats (CSV, Bases de données, XML, JSON).
- Enrichissement de données par croisement.
- Programmation Java de traitements de transformation et croisement personnalisés.

Documentation:

- https://help.talend.com/

Installation de l'environnement de travail

Manipulation 1: Vérifier la version installée de Java, le cas échéant, avec les commandes : *java -version* et *javac -version*

Manipulation 2: Si au moins une des deux commandes ne fonctionne pas, installer Java/JDK (version 15 à 19) en le téléchargeant à partir de : https://www.oracle.com/java/technologies/javase/jdk15-archive-downloads.html

Manipulation 3: Télécharger l'outil ETL Talend *Open Studio for Data Integration* (version 8) à partir de : https://sourceforge.net/projects/talend-studio/files/Talend%20Open%20Studio/8.0.1M12/

Manipulation 4: Le décompresser avec l'outil *7zip* ou *Winrar* (ne pas utiliser le décompresseur de Windows).

Manipulation 5: Démarrer Talend et créer le workspace *tp1-di*.

Partie 1 : Traitement de sources uniques

1. Chargement de fichiers de données en CSV et Excel

Manipulation 6: Ajouter un fichier csv dans le dépôt de Talend.

Manipulation 7: Charger les données du fichier à l'aide du composant tFileInputDelimited.

Manipulation 8: Afficher les données avec tLogRow.

Manipulation 9: Enregistrer les données en résultat dans un fichier csv puis Excel.

Manipulation 10: Refaire le même travail à partir d'un ficher Excel à l'aide de tFileInputExcel.

2. Traitement de données

Manipulation 11: Filtrer les données (par civilité, par nom et/ou par année de naissance par exemple, utiliser la fonction de Talend *TalendDate.getPartOfDate* pour extraire les partie d'une date complète) et afficher le résultat. tFilterRow.

Manipulation 12: Dé-doublonner les données suivant une clé de gestion puis enregistrer le résultat dans un fichier que vous utiliserai pour la suite du TP. Utiliser tUniqRow et *tFileOutputxxx*.

Manipulation 13: Transformer les nom et prénom en majuscules. Utiliser tMap.

Manipulation 14: Ajouter un champ contenant la concaténation du nom et prénom. Utiliser tMap.

Manipulation 15: Calculer l'âge des personnes à partir de leur date de naissance. Utiliser tMap.

Manipulation 16: Remplacer les valeurs numériques de la civilité par les lettres correspondantes (1 par M, 2 par Mme et 3 par Mlle).

Manipulation 17: Faire des calculs d'agrégation (nombre de personne par civilité/âge par exemple). Utiliser tAggregateRow.

Manipulation 18: Ajouter une boite de dialogue permettant à l'utilisateur la saisie de critères de sélection. Utiliser tMsgBox (les saisies sont récupérées dans la variable globale NomComposant_RESULT à récupérer de la structure *globalMap*).

3. Chargement de base de données

Manipulation 19: Ajouter une connexion à une base de données MySQL dans le dépôt du projet.

Manipulation 20: Charger les données du fichier à l'aide du composant tMySQLInput.

Manipulation 21: Afficher les données à l'aide du composant tLogRow.
Manipulation 22: Filtrer les données et afficher le résultat.

Manipulation 23: Enregistrer les données dans un fichier (csv, excel, etc.).

4. Mises à jour de bases de données

Manipulation 24: Enregistrer les données du fichier CSV plus haut dans la base de données.

Utiliser tMySQLOutput.

Manipulation 25: Mettre à jour les informations des clients (adresse, téléphone, email, etc.) enregistrés dans la base de données (composant tMySQLOutput). Les informations à jour sont fournies dans un fichier CSV (et/ou XML).

5. Chargement de données XML et JSON

Manipulation 26: Ajouter un fichier XML dans le Repository de Talend.

Manipulation 27: Charger les données XML à l'aide du composant tFileInputXML.

Manipulation 28: Afficher les données sous forme de table avec tLogRow.
Manipulation 29: Transformer les noms et prénoms en majuscules (attention aux valeurs nulles, faire le test row.nom==null ? "" : row.nom.toUpperCase()).
Manipulation 30: Calculer l'âge des personnes à partir de leur date de naissance et l'ajouter en champ supplémentaire.
Manipulation 31: Filtrer les données (par année de naissance par exemple).

Manipulation 32: Faire le même travail avec des données JSON avec tFileInputJSON.

Partie 2 : Croisement de données multisources

6. Croisement de données CSV

Manipulation 33: Faire le croisement des deux fichiers CSV avec tMap.
Manipulation 34: Afficher le résultat du croisement avec tLogRow puis l'enregistrer dans un fichier sur disque avec tFileOutpuDelimited.

7. Croisement de bases de données

Manipulation 35: Injecter deux bases de données (de clients) avec tMySQLInput.
Manipulation 36: Faire le croisement des deux bases avec tMap.
Manipulation 37: Afficher le résultat du croisement avec tLogRow puis l'enregistrer dans un nouveau fichier.

8. Croisement de données CSV, XML et JSON

Manipulation 38: Insérer un fichier XML dans le dépôt du projet.
Manipulation 39: Lire les données du fichier XML à l'aide de tFileInputXML.

Manipulation 40: Faire le croisement de ces données XML avec les données CSV puis avec une base de données avec tMap.
Manipulation 41: Afficher le résultat du croisement avec tLogRow puis l'enregistrer dans un nouveau fichier.

Manipulation 42: Croiser des données JSON avec XML et afficher/enregistrer le résultat en XML puis en JSON.

Partie 3 : Croisement de données internes avec des données externes

9. Ingestion de données de services Web REST et SOAP

Manipulation 43: Collecter les données (JSON et XML) de services Web REST GetGeoApi (Currency Convertor : https://currency.getgeoapi.com) et API JCDecaux (vélos en libre de service : https://developer.jcdecaux.com) à l'aide de tREST et les afficher avec tLogRow.

Manipulation 44: Utiliser tExtractJSONFields ou tExtractXMLField selon le format des réponses pour extraire les données JSON ou XML et les enregistrer dans un fichier texte (au format CSV) puis dans une base de données MySQL.

Manipulation 45: A partir du fichier SalariesProf.xlsx, récupérer les taux de change des devises du fichier vers l'Euro afin d'harmoniser les salaires du fichier.

Manipulation 46: On voudrait récupérer les stations Velib de plusieurs villes fournies dans un fichier csv (ou récupérées par l'API REST). Utiliser tFlowToIterate pour réexécuter le service Web tREST pour chaque ligne récupérée du fichier csv (penser à reparamétrer tREST).

10. Croisement de données de services Web

Manipulation 47: Utiliser les données issues des services Web pour enrichir les bases internes avec tMap.

Partie 4 : Personnalisation des traitements avec Java

11. Utilisation de composants de développement Java

Manipulation 48: Utiliser le composant tJavaRow pour parcourir un fichier csv/xml de clients et concatène le nom et prénom dans un seul attribut (après avoir transformé le nom en majuscules).

Manipulation 49: Utiliser le composant tJavaRow pour Programmation de routines Java dans Talend

Manipulation 50: Dans une nouvelle routine Java, écrire une méthode de calcul d'âge à partir d'une date de naissance.
Manipulation 51: Utiliser cette méthode dans un tMap qui croise des personnes par âge.

Manipulation 52: Ecrire une méthode permettant de concaténer le nom et prénom d'une personne en une seule chaîne de caractères.
Manipulation 53: Utiliser cette méthode dans un tMap.

Manipulation 54: Créer une nouvelle routine et programmer la méthode *double operation (v1, v2, op)* permettant d'évaluer l'opération en paramètre.
Manipulation 55: Utiliser cette méthode dans un tMap pour évaluer les opérations d'un fichier csv/xml.