Data Integration, Talend

June 22, 2023

Part 1

Introduction

Au cours de cet exercice, nous allons travailler avec une base de données de films. Nous disposons de 3 sources différentes:

- moviesRelease : une base de données indiquant le titre, l'année de sortie, une page du film ainsi qu'un résumé du film (le fichier moviesRelease.sql permet de créer cette base de données)
- moviesIncome : un fichier csv indiquant le budget et le revenu de chaque film
- moviesRatings : un fichier xml avec des indicateurs de popularité du film

Dans moviesRelease et moviesCincome, les films sont identifiés par un id, que l'on suppose bien formé pour l'ensemble : chaque film a le même id dans les deux fichiers. moviesRating identifie ses films par leur titre (correspondant au titre de sortie).

Lors de ce tp, vous allez devoir créer un ou plusieurs jobs talend afin d'extraire et de transformer ces données. L'ensemble des jobs terminés, vous devez les soumettre sur blackboard (pour cela, faites un clic droit sur les jobs > exporter). Vous devrez également fournir dans une archive les fichiers demandés.

Il est demandé de donner des noms explicites à toutes les variables (tables, colonnes, page de sortie etc.). Il est possible de modifier le nom de l'objet dans l'onglet "vue" une fois un composant sélectionné.

Part 2

Travail à faire

Question 1

a) Commencez par récupérer les données, pour cela, créez trois composants, un pour chaque source.

b) Vérifiez que les sources sont correctes avec un affichage (que vous désactivez ensuite).

c) Créez une table unique avec tous les champs des différents fichiers. Vous utiliserez cette table comme base pour la suite de l'exercice.

On ne souhaite désormais travailler qu'avec les films plus récents que 2000. Afin de comparer des dates avec Talend (et en Java de manière générale), on ne peut pas juste utiliser un comparateur classique. La méthode permettant cette opération avec talend est : TalendDate.compareDate([DATE1], [DATE2]) >= 0

Pour créer une Date à partir d'une chaine de caractères, on utilise : TalendDate.parseDate("dd-MM-yyyy", [DATE_STRING])

Question 2

a) Filtrez les données afin de ne garder que les films suffisamment récents. Exportez les films plus vieux dans un fichier csv

Un premier client souhaite étudier les liens entre budget, revenus et popularité des films. Ses outils sont adaptés au traitement de fichiers csv et excel.

Question 3

a) Fournissez dans un fichier au format adapté les données nécessaires. Le fichier doit être trié par titre de film.

Un deuxième client souhaite travailler sur la relation entre budget, revenue et date de sortie du film. Il ne peut travailler que sur des données complètes et dans fichier csv dont le séparateur est "„,". Le fichier doit être trié par année de sortie puis par titre de film.

Question 4

a) Produisez le fichier désiré.

Enfin, la source vous ayant fourni moviesRatings aimerait avoir un fichier similaire mais enrichi avec le nom des films correspondant.

Question 5

a) Produisez le fichier désiré.

Le deuxième client souhaiterait désormais avoir un document excel avec plusieurs pages : la première regroupe les mêmes informations que précédemment. La deuxième présente les revenus et budgets cumulés par an ainsi que le nombre de films sortis cette année-là. Cette page est triée par an. La troisième affiche le gain (ou perte) de chaque film. Cette page est triée par gain.

Question 6

a) Mettez à jour votre projet afin de fournir le document demandé.