Rakib SHEIKH
Travaux pratiques : TRDE703 Intégration de données
rakib.sheikh@cyu.fr

TP : Intégration de données

Le sujet est cours de rédaction et des modifications peuvent encore arriver en cours de route.

La date du rendu est fixé au 24 février 2025 à 6h00 du matin.

Outils pour ce TP

Les outils d'ETL que vous utiliserez sera au choix parmi les outils suivants :

- **Talend** (Aujourd'hui nommé Taxalie Data Integration) : https://deilink.fr/#/download
  - L'outil est multi-plateforme.
- **Alteryx** (https://www.alteryx.com/fr/sparked/learning-programs)
  - Uniquement disponible sous Windows
- **Knime** : https://www.knime.com/downloads Ne remplissez pas le formulaire, cliquez directement sur Download
  - Multi-plateforme
- **Tableau Prep** : https://www.tableau.com/academic/students
  - Pas disponible sous Linux
- **Dataiku** (https://www.dataiku.com/product/get-started/) Descendez dans la page pour « Free Edition »
  - L'outil est multi-plateforme
- **amphi.ai** : A titre expérimentale
  - L'outil est multiplateforme.

Le choix de la base de donnée cible est **libre**. Recommendation personnel : postgresql.

Rendu :

Rapport sur l'ensemble des exercices avec des captures d'écran sur l'ensemble des jobs et du résultat finale. Votre rapport doit détailler l'ensemble des étapes réalisés pour arriver à votre résultat.

- Composition de groupe maximale : 4 personnes

Exercice 1 : Un peu de théorie pour débuter

Pour cette excercice, vous pouvez utiliser l'outil suivant pour modéliser vos tables : https://www.drawdb.app/

Nous disposons de deux sources de données relatives à l'achats d'articles

- Source 1

```
Produit(numP, nomP, prixP, délaiLiv, stock)
Fournisseur(numF, nomF, localisation, remise) nomF est unique
Localisation de la forme (ville, province, pays)
Achat(noAchat, dateAchat, moyenPaiment, numF#)
AchatUnit(noAchat#, numP#, qte, prixUnit)
```

- Source 2

```
ProdAchats(codeProd, dateAchat, descProd, catProd, nomFour, qte, dispo, prixUnit, montant)
```

On considère qu'il y a 1000 produits et 10 fournisseurs. Chaque produit est fourni par exactement un fournisseur et chaque achat comporte 5 produits. On considère qu'il a 20 achats par jour.

Un même produit ne peut être renseigné que dans l'une des deux sources et jamais dans les deux à la fois.

Le but est de suivre l'évolution de l'état des stocks et des montants dépensés dans le temps.

Pour chaque période de temps et chaque zone géographique mesurer les informations suivantes :
- la quantité moyenne des stocks d'un produit donné
- la somme total gagnée par chaque fournisseur
- le pourcentage du prix de chaque produit dans les achats effectués
- le taux de participation de chaque fournisseur, en terme de montant gagné, dans les achats effectués

La granularité temporelle comprend la semaine, le mois, le trimestre, et l'année. La granularité spatiale comprend la ville, la province et le pays.

Pour ce faire, il faudra :
1. Identifier les dimensions utiles au problème. Expliquer comment obtenir chaque dimension à partir des données sources. Indiquer quelles dimensions sont hiérarchiques et expliciter leur hiérarchie.
2. Définir les mesures et les fonctions d'agrégation associées
3. Déduisez le schéma multidimensionnel en considérant que la table de faits est normalisée et que les tables décrivant les dimensions sont dé-normalisées. Discuter le choix de la granularité choisie pour les dimensions hiérarchiques.
4. On considère les statistiques suivantes. Estimer la taille, en nombre de ligne, de la table de faits en considérant que les données portent sur une année.
5. On voudrait revoir la conception pour obtenir la quantité totale de produit vendus pour chaque catégorie. Quelle modification apporter et est-ce que peut poser un problème de cohérence et quelle solution suggérer ?

Exercice 2 – Outil d'extraction, transformation chargement (ETL)

L'objectif de cette exercice est de vous introduire à la constructions de pipelines de données depuis un outil d'ETL.

- Voici la source de donnée à intégrer :
  - fichier : `movies.csv`
  - Datacard : https://www.kaggle.com/datasets/bharatnatrayn/movies-dataset-for-feature-extracion-prediction

Réalisez toute les transformations nécessaire afin d'arriver un dataset pouvant être intégrés dans votre base de données postgres. Votre donnée finale devra être sous forme structuré (c'est à dire relationnel)

Dans le rapport, vous justifiez l'ensemble de vos transformation que vous avez appliqué.

Exercice 3 – Mise en oeuvre d'un cube OLAP

L'exercice est finalisé, mais n'est pas encore testé en son intégralité

Nous allons utiliser l'outil Atoti.io afin de modéliser notre premier cube OLAP.

1. Installez Atoti avec la ligne de commande suivante

```
pip install "atoti[jupyterlab]"
```

2. Nous allons générer un dossier tutoriel que vous allez pouvoir suivre pas à pas pour votre propre données. Pour cela, les données à utiliser sera ceux présenté lors du TP de Visualisation de données.
   - Générez le document du tutoriel avec la commande suivante :

```
python -m atoti.copy_tutorial tutorial
jupyter lab
```

3. Imitez le tutoriel sur les données de la journée 1 du cours de Visualisation de données.

Exercice 4 – Intégration des données non-tabulaire

L'exercice est en cours de rédaction

Partie 1 : Extraction

Jusqu'à présent, nous nous sommes contenté de travailler avec des données qui sont principalement représentés sous la forme de données tabulaire. Comme dans le monde du Big Data, les données peuvent admettre une certaine variété. C'est ce que nous allons voir à travers de cette exercice qui va vous faire sortir de votre zone de comfort.

Notre but : Récupérer des données semi-structurés afin de les transformer en des données structurés. Tout en respectant les propriétés d'un cube OLAP

La difficulté de cette exercice réside sur le fait qu'il y a plusieurs API de données de sources différentes à utiliser.

- Source de données 1 : hhttps://portail-api.insee.fr/catalog/api/2ba0e549-5587-3ef1-9082-99cd865de66f
- Source de données 2 : https://www.data.gouv.fr/fr/dataservices/api-sirene-open-data/
- Source de données 3 : https://adresse.data.gouv.fr/outils

Objectif à terme : Avoir une base de données regroupant les offres d'emploi, les entrerises associés, ainsi que sa localisation

Partie 2 : Transformation

- Appliquez les transformation de données nécessaire afin qu'elles puissent être intégrés dans votre base de donnée cible.
- Appliquez égalemnt la normalisation sous la forme d'un modèle en étoile à minima, et si les données le rend possible, un modèle en flocon.
- N'oubliez pas que votre transformation de données doit répondre aux propriétés de cubes OLAP.
  - A l'aide de vos connaissances acquises lors de l'exercice 3, appliquez les propriétés des cubes OLAP. Le choix des dimensions du cube est libre.

Partie 3 : Load

- Maintenant que votre job pour la transformation de données est terminé, il ne reste plus qu'à charger le résultat de votre traitement de données vers la base de donnée cible de votre choix.

Partie 4 : Une petite visualisation de donnée ? (Optionel)

- Afin de bien marquer le coup de ce TP, proposez une petite visualisation rapide des données finales que vous avez transformés.
  - Ici, vous utiliserez un outil de visualisation de données parmis la liste suivante : PowerBI, Tableau, Apache Superset