Republic of Tunisia
Ministry of Higher Education and Scientific Research
University of Tunis
National School of Engineering of Tunis

Ecole Nationale Supérieure d'Ingénieurs de Tunis

Engineer in Applied mathematics and modeling
Current academic year 2023/2024

Haythem Boumellassa **3 ème Gmam**

Rapport de TP : ETL avec Apache NiFi

Objectif

L'objectif de ce TP est de se familiariser avec les formats, traitements et manipulations possibles des données dans un contexte de projet BI/Big Data en utilisant Apache NiFi. Les étapes énumérées permettent de comprendre les fonctionnalités essentielles de l'outil.

Préparation de l'environnement

Arborescence

- **Création de l'arborescence suivante :**
- **/home/<user>/gmam/SecondSession/data/input**
- **Téléchargement du fichier turnover.xls et copie dans le répertoire input.**

Lancement d'Apache NiFi

- **Démarrage d'Apache NiFi : -E ./nifi.sh start).**
- **Accès à l'interface web : http://<adresse_ip>:8080/nifi.**

Construction du DataFlow

1. Création d'un Process Group

- **Nom : GMAM-RELIABILITY-TP1.**

2. Lecture du fichier

- **Processeur utilisé : GetFile**
  - **Configuration :**
    - **Input Directory : /home/<user>/gmam/SecondSession/data/input**
    - **File Filter : turnover.xls**

3. Ajout d'une colonne avec la date et l'heure

- **Processeur utilisé : UpdateAttribute**
  - **Expression Language :**
  - **${now():format('yyyy-MM-dd HH:mm:ss')}**
  - **Nom de la colonne ajoutée : ingestion_timestamp.**

4. Conversion en format Parquet

- **Processeur utilisé : ConvertRecord**
  - **Configuration :**
    - **Record Reader : AvroReader**
    - **Record Writer : ParquetRecordSetWriter.**

5. Compression Snappy/GZIP

- **Processeur utilisé : CompressContent**
  - **Compression Format : Snappy ou GZIP.**

6. Partition par la colonne "industry"

- **Processeur utilisé : PartitionRecord**
  - **Configuration :**
    - **Partition Field : industry**
    - **Résultat attendu : 16 FlowFiles, chacun correspondant à une valeur unique de la colonne "industry".**

7. Mise à jour de l'attribut "filename"

- **Processeur utilisé : UpdateAttribute**
  - **Expression Language :**
  - **${industry}_${UUID}**

8. Chargement des fichiers dans les répertoires cibles

- **Processeur utilisé : PutFile**
  - **Output Directory :**
  - **/home/<user>/gmam/SecondSession/data/output/${industry}**

9. Export du DataFlow en format JSON

- **Action : Création d'un template (à partir de l'interface NiFi) et export au format JSON.**
- **Dépôt GitHub :**

- **Création d'un dépôt, upload du fichier JSON et XML**

Définitions des notions rencontrées

Partitioning

- **Technique de division des données en plusieurs segments selon des clés (ex : colonne "industry").**

Compression (Snappy/GZIP)

- **Snappy : Compression rapide et efficace, adaptée aux systèmes Big Data.**
- **GZIP : Compression plus lente mais avec un taux de compression élevé.**

Apache NiFi Expression Language

- **Langage permettant de manipuler dynamiquement des attributs et des contenus au sein des processeurs.**

Optimisations possibles

1. **Automatisation du déploiement :**
   - **Utilisation de scripts pour créer et déployer automatiquement les DataFlows.**
2. **Amélioration des performances :**
   - **Augmentation du parallélisme (threads) dans les processeurs pour accélérer le traitement.**
3. **Audit et supervision :**
   - **Ajout de processeurs pour surveiller et logger les échecs.**

Commandes Git utilisées

1. **Initialisation :**
2. **git init**
3. **Ajout et commit :**
4. **git add .**
5. **git commit -m "Ajout du template NiFi pour le TP1"**
6. **Création d'un dépôt distant :**
7. **git remote add origin <https://github.com/Beru47/ETL-with-nifi>**
8. **Envoi des modifications :**
9. **git push -u origin main**

Lien GitHub

[Template Github repository]