

TD Migration de données simple

Niveau 1

Utilisation avec DBeaver

Nous avons utilisé des conteneurs pour créer la base de donnée MySQL et PostgreSQL et récupéré leur adresse ip :

```
louxif@louxif-Latitude-5590:~$ docker inspect -f '{{range .NetworkSettings.Networks}}{{.IPAddress}}{{end}}' mon_mysql
172.17.0.2
louxif@louxif-Latitude-5590:~$ docker inspect -f '{{range .NetworkSettings.Networks}}{{.IPAddress}}{{end}}' mon_postgre
172.17.0.3
```

Nous avons rempli rapidement la base de donnée MySQL avec quelques données via ces commandes :

```
CREATE DATABASE test;
USE test;
CREATE TABLE utilisateurs (
  id INT AUTO_INCREMENT PRIMARY KEY,
  nom VARCHAR(50),
  prenom VARCHAR(50),
  email VARCHAR(100) UNIQUE,
  age INT,
  genre ENUM('M', 'F', 'Autre'),
  adresse VARCHAR(255),
  code_postal VARCHAR(20),
  ville VARCHAR(100),
  pays VARCHAR(100),
  date_inscription DATETIME DEFAULT CURRENT_TIMESTAMP
);
INSERT INTO utilisateurs (nom, prenom, email, age, genre, adresse, code_postal, ville, pays) VALUES
('Doe', 'John', 'john.doe@example.com', 30, 'M', '123 rue de la Ville', '75001', 'Paris', 'France'),
('Smith', 'Emma', 'emma.smith@example.com', 25, 'F', '456 Avenue du Soleil', '69001', 'Lyon', 'France'),
('Garcia', 'Maria', 'maria.garcia@example.com', 35, 'F', '789 Boulevard des Fleurs', '31000', 'Toulouse', 'France'),
('Jones', 'Michael', 'michael.jones@example.com', 40, 'M', '987 Rue de la Mer', '33000', 'Bordeaux', 'France'),
('Chen', 'Li', 'li.chen@example.com', 28, 'F', '321 Street of Flowers', '100001', 'Shanghai', 'China'),
('Kumar', 'Raj', 'raj.kumar@example.com', 33, 'M', '456 Main Street', '560001', 'Bangalore', 'India');
```

Connections a DBeaver :

Configuration de la connexion "localhost"

MySQL paramètres de connexion

Paramètres de connexion

Initialisation
Commandes Shell
Identification du client
Transactions
Général
Métadonnées
Gestion des erreurs
Data Transfer
▶ Jeux de résultats
▶ Editeur SQL

Général Propriétés du pilote SSH SSL + Network configurations...

Server

Connect by: ☐ Host ☒ URL

URL: jdbc:mysql://172.17.0.2:3306/test?allowPublicKeyRetrieval=true

Server Host: localhost Port: 3306

Database:

Authentication (Database Native)

Nom d'utilisateur : root

Mot de passe : ☒ Enregistrer les mots de passe

Advanced

Server Time Zone: Auto-detect ▼

Local Client: ▼

① [Vous pouvez utiliser des variables dans les paramètres de connexion.](#)

Driver name: MySQL Driver Settings License du pilote

Test de la connexion ... Annuler OK

Configuration de la connexion "postgres"

Paramètres de connexion
PostgreSQL paramètres de connexion

▼ Paramètres de connexion

- Initialization
- Commandes Shell
- Identification du client
- Transactions
- Général
- Métadonnées
- Gestion des erreurs
- Data Transfer
- Jeux de résultats
- Editeur SQL

Général PostgreSQL Propriétés du pilote SSH SSL + Network configurations...

Server

Connect by: ☐ Host ☒ URL

URL: jdbc:postgresql://172.17.0.3:5432/postgres

Host: localhost Port: 5432

Database: postgres ☐ Show all databases

Authentification

Authentification: Database Native

Nom d'utilisateur : postgres

Mot de passe : ☒ Enregistrer les mots de passe

Advanced

Session role: Local Client:

[Vous pouvez utiliser des variables dans les paramètres de connexion.](#)

Driver name: PostgreSQL Driver Settings License du pilote

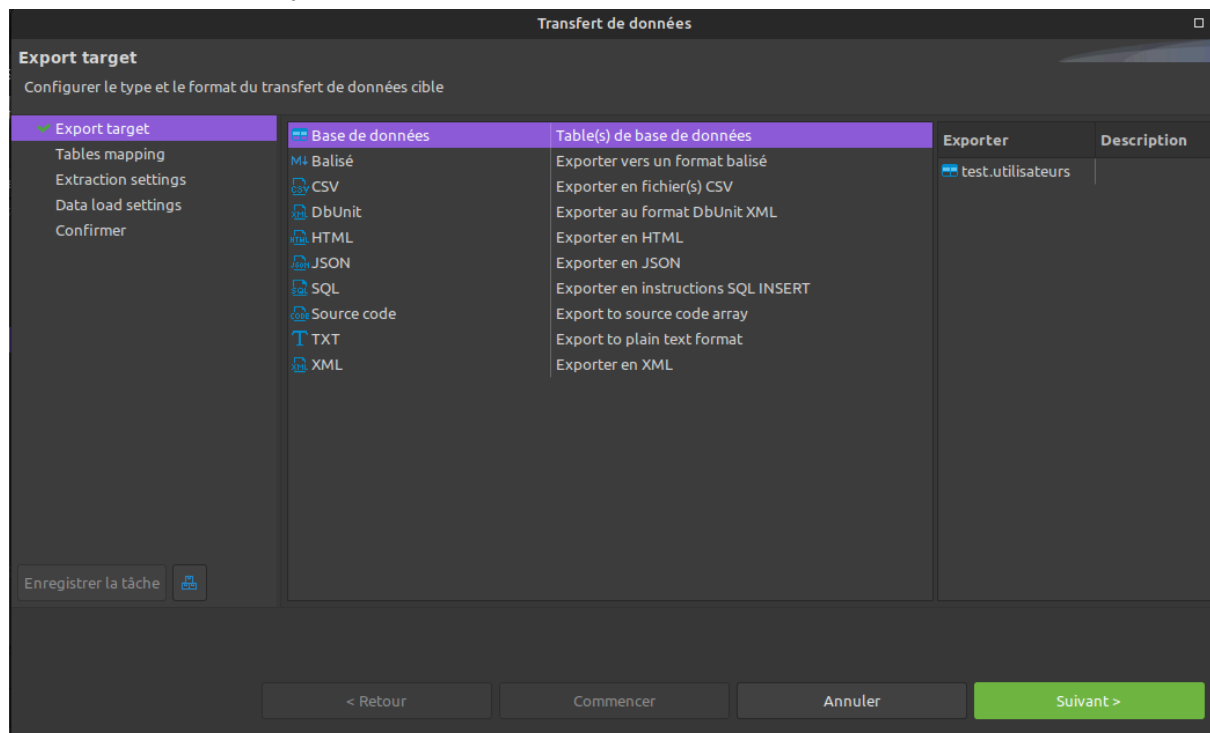
Test de la connexion ... Annuler OK

Création de la table utilisateurs dans Postgre :

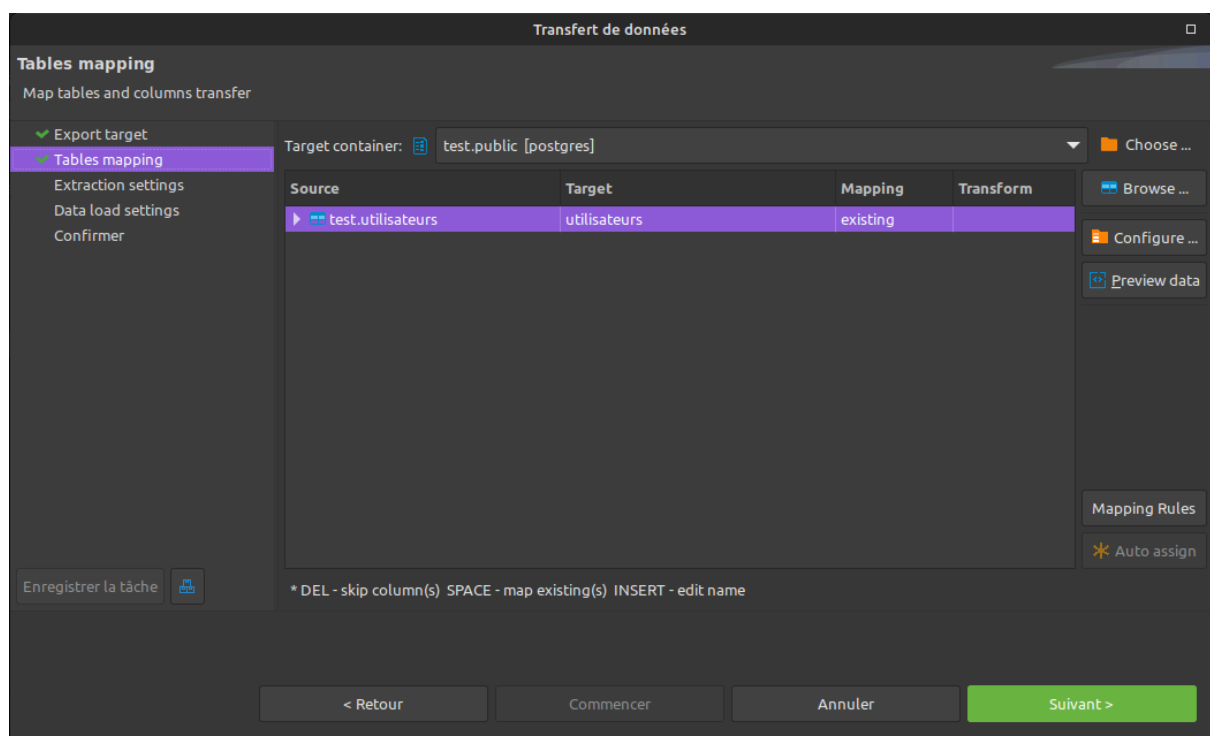
```
CREATE TABLE utilisateurs (
  id SERIAL PRIMARY KEY,
  nom VARCHAR(50),
  prenom VARCHAR(50),
  email VARCHAR(100) UNIQUE,
  age INT,
  genre CHAR(1),
  adresse VARCHAR(255),
  code_postal VARCHAR(20),
  ville VARCHAR(100),
  pays VARCHAR(100),
  date_inscription TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

Migration des données :

Nous avons choisi le type de transfert, ici Table de base de donnée :



Ici nous avons choisi la table cible :



Sur les deux prochaines étapes, nous n'avons rien eu a modifier :

Transfert de données

Extraction settings

Database table(s) extraction settings

- ✓ Export target
- ✓ Tables mapping
- ✓ Extraction settings
- Data load settings
- Confirmer

Progression

Tâches (threads) maximales:

Type d'extraction: Requête unique ▼

Taille des segments:

☐ Ouvrir connexion(s)

☐ Sélectionner le nombre de lignes

Fetch size:

Enregistrer la tâche

< Retour
Commencer
Annuler
Suivant >

Transfert de données

Data load settings

Configuration of table data load

- ✓ Export target
- ✓ Tables mapping
- ✓ Extraction settings
- ✓ Data load settings
- Confirmer

Data load

☒ Transfer auto-generated columns

☐ Truncate target table(s) before load

☐ Disable referential integrity checks during the transfer

Replace method: <None> ▼

[Replace/Ignore method documentation](#)

General

☒ Open table editor on finish

☒ Show finish message

Performance

☒ Ouvrir connexion(s)

☒ Use transactions

Do Commit after row insert:

☐ Use multi-row value insert

☐ Skip bind values during insert

☐ Disable batches

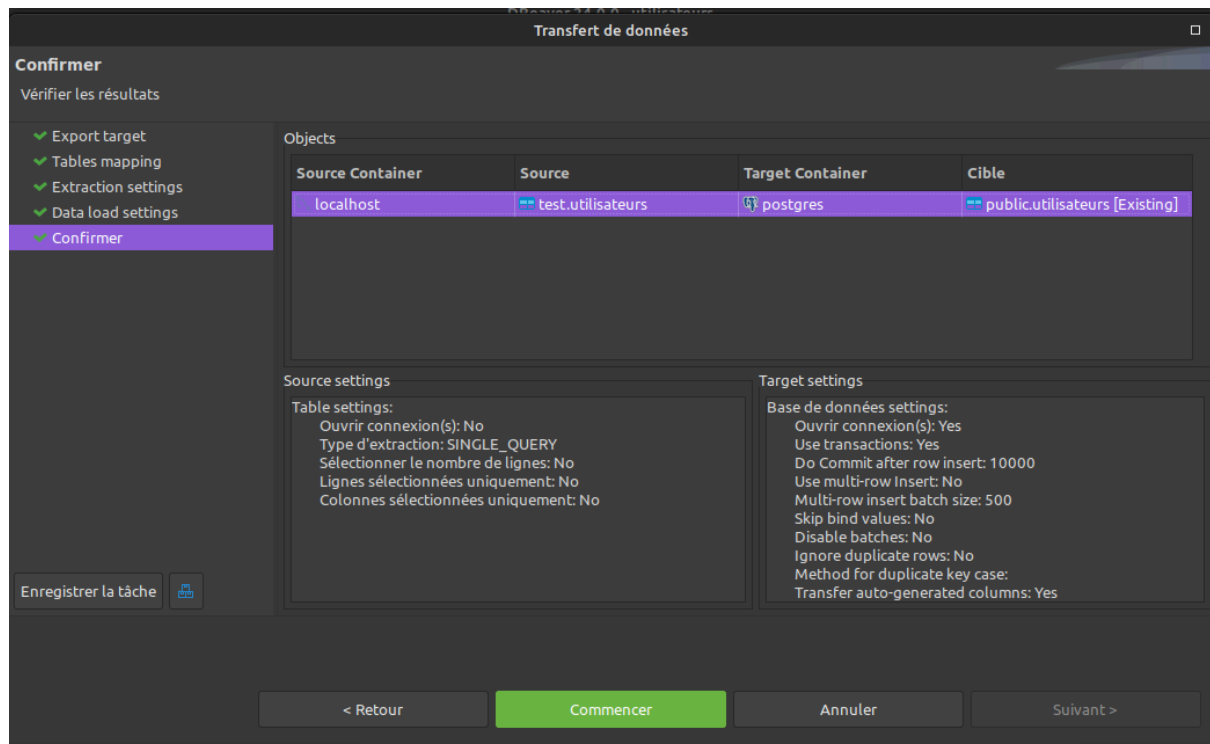
☐ Ignore duplicate rows errors

☐ Use bulk load

Enregistrer la tâche

< Retour
Commencer
Annuler
Suivant >

Voilà le récapitulatif de la migration :



Nous pouvons voir nos données qui sont arrivées sur la bdd Postgre :

