



**MINISTÈRE
DE LA TRANSITION
ÉCOLOGIQUE
ET DE LA COHÉSION
DES TERRITOIRES**

*Liberté
Égalité
Fraternité*

Offre Eole 2.8

Variante géomatique

Migration des données



Historique des versions du document

Version	Date	Commentaire
Vo	02/2019	Document Ch. Bocquet relatif à la migration de données pour Eole 2.7
V1	11/2022	Migration de données vers Eole 2.8

Affaire suivie par

Philippe LOUSTAUNAU – DNUM/UNI/DRC
Tél. :
Courriel :

Alain FERRATON – DNUM/MSP/DS/GSG
Tél. :
Courriel :

Rédacteur

Philippe LOUSTAUNAU – DNUM/UNI/DRC à partir du document rédigé en 2019 par Ch. Bocquet

Relecteur

Alain FERRATON – DNUM/MSP/DS/GSG

Vincent CHAVANON – DNUM/UNI/DETN/GSIL/PPSRL

Référence(s) intranet

<http://geoinformations.metier.e2.rie.gouv.fr/postgresql-postgis-r702.html>

Introduction

Le présent document traite uniquement de la migration des bases de données vers un serveur SETH V2.8.x en variante géomatique. Ce document ne concerne **pas** les serveurs variante geobase.

Il n'est pas prévu de migration massive et obligatoire des serveurs. Il appartient à chaque service de choisir le moment opportun pour migrer son ou ses serveurs.

Il est toutefois rappelé que les versions Eole 2.6 ne sont plus supportées depuis mi-2021 et que le support des Eole 2.7.x s'arrêtera en avril 2023. L'offre Eole 2.8 variante géomatique (offre PostgreSQL) sera accompagnée jusqu'en avril 2025 (cf. [Nouvelles orientations pour Eole – Avril 2022](#))

Cette variante intègre PostgreSQL 12, Postgis 3 et pgAdmin4 serveur v6.7. Elle bénéficie de la distribution et mise à jour automatique des extensions métiers Asgard et bientôt Plume destinées à faciliter l'utilisation des données.

Elle s'adresse à tous les services du MTE ainsi qu'aux DDT(M).

Migration d'un serveur Seth 2.x vers la version 2.8

Il existe différentes possibilités et différents cas de figure peuvent se présenter dans les services pour migrer vers un serveur Eole 2.8. Volontairement, ce document se limite à deux cas de figure :

- Installation de la version 2.8 sur la même machine, à la place de l'ancien système. Cette stratégie induit nécessairement une interruption de service mais ne nécessite pas obligatoirement un second serveur disponible. On pourra utilement consulter [la fiche relative à la réinstallation avec préservation de données](#).
- Installation de la version 2.8 sur une autre machine, à côté du serveur à migrer. Cette option évite toute interruption de service. Une fois le nouveau serveur opérationnel, on peut stopper l'ancien et réattribuer son adresse IP au nouveau serveur en modifiant la conf zephir et en provoquant une mise à jour.

Pour rappel : Lors de l'installation du serveur, il est **OBLIGATOIRE** de lancer la commande `Maj-Auto -i -S debmiroir-02.eole.e2.rie.gouv.fr -U debmiroir-02.eole.e2.rie.gouv.fr -V debmiroir-02.eole.e2.rie.gouv.fr AVANT enregistrement_zephir`.

Pour installer PostgreSQL depuis le Zephir, on se reportera à la [fiche de procédure qui explique également comment initialiser le rôle de connexion](#) vers la base de données 'adl' créée par défaut.

Une fois le serveur installé, il est nécessaire d'initialiser le compte de connexion à pgAdmin4 serveur, outil de gestion de bases de données (cf. [fiche sur pgAdmin4 serveur](#))

Migration des données

Deux stratégies ont possibles pour récupérer les informations issues de bases de données d'un serveur migré :

- Une migration complète qui permet de restaurer d'un seul coup l'ensemble des bases, rôles et droits
- Une migration base par base qui permet de choisir les bases à restaurer, voire de restaurer des bases issues d'un autre serveur que celui migré.

I. Méthode 1 : restauration complète.

Cette méthode permet de restaurer d'un seul coup l'ensemble des bases, des rôles, et de leurs droits. Toutes les commandes sont à exécuter en tant que root.

(A) 3.1 - Première étape : sauvegarde

Dans le répertoire /home/backup/postgresql, un certain nombre de fichiers sont générés chaque jour en fonction des choix de sauvegarde faits par l'administrateur des données.

Il s'agit de dumpall.dmp qui correspond à la sauvegarde de tous les rôles avec leurs mots de passe et d'un fichier avec une extension .backup par base de données sauvegardée et qui contient **les structures et les données des différentes bases sauvegardées pour les schémas sauvegardés**.

Même si l'idée ici n'est pas d'utiliser ces fichiers pour restaurer les données base par base (ce qui sera décrit dans la méthode 2), il est judicieux de vérifier que ces fichiers sont à jour (contrôle de la date) et de les sauvegarder par ailleurs par mesure de précaution.

Ensuite, il faut générer un fichier correspondant à la sauvegarde de la totalité du cluster. Dans le cas de bases volumineuses, il faut s'assurer de disposer d'assez de place sur le serveur avant de lancer la commande.

Dans tous les cas, il faut être certain que pendant cette sauvegarde, aucune activité de modification des bases ne peut avoir lieu. Cela peut-être fait en modifiant postgresql.conf.

Le paramètre listen_addresses permet de déterminer les adresses IP autorisées à accéder au serveur. Il suffit de limiter l'accès à localhost (listen_addresses='localhost'), sauvegarder postgresql.conf et redémarrer postgresql :

```
/etc/init.d/postgresql stop
```

```
/etc/init.d/postgresql start
```

Ensuite, vérifier que l'on est dans le répertoire /root et qu'un fichier .pgpass est bien présent puis lancer la commande :

```
pg_dumpall -U postgres -f /home/backup/postgresql/tout.dmp
```

Au moyen d'un éditeur de texte (nano par exemple), ouvrir le fichier /home/backup/postgresql/tout.dmp et **supprimer** les deux lignes concernant la création du rôle postgresql en début de fichier :

```
CREATE ROLE postgres;  
ALTER ROLE postgres WITH SUPERUSER INHERIT CREATEROLE CREATEDB LOGIN REPLICATION  
BYPASSRLS PASSWORD 'xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx';
```

Récupérer ce fichier.

(B) 3.2 - Seconde étape : restauration

Copier le fichier tout.dmp récupéré à la première étape après l'installation du nouveau système, l'enregistrement zephir, l'instanciation avec la bonne variante et toutes les mises à jour, dans le répertoire /home/backup/postgresql/

En étant connecté en root dans le répertoire /root, après avoir vérifié la présence du fichier .pgpass, lancer la commande :

```
psql -U postgres -d adl -f /home/backup/postgresql/tout.dmp
```

La restauration doit se dérouler en entier sans erreur. Vérifier que tout a bien été récupéré (bases, rôles, droits) en se connectant au nouveau serveur.

II. Méthode 2 : restauration base par base.

Pour différentes raisons, il peut être opportun de procéder à une restauration base par base. Par exemple :

- Certaines bases n'ont plus d'utilité ou sont des référentiels non produits localement qui seront réintégrés ensuite
- La volumétrie totale est très grosse et une restauration en plusieurs étapes est plus facile
- ...

(C) 4.1 - Première étape : sauvegarde

Dans le répertoire /home/backup/postgresql, un certain nombre de fichiers sont générés chaque jour en fonction des choix de sauvegarde faits par l'administrateur des données. Il s'agit de dumpall.dmp qui correspond à la sauvegarde de tous les rôles avec leurs mots de passe et d'un fichier avec une extension .backup par base de données sauvegardée et qui contient **les structures et les données des différentes bases sauvegardées pour les schémas sauvegardés**.

Il faut commencer par vérifier la stratégie de sauvegarde pour s'assurer que tout ce qui doit être migré est sauvegardé. Si nécessaire, la modifier en intervenant dans la base « adl » et dans ce cas, attendre le lendemain (cf. [Guide offre éolisée PostgreSQL §2.2](#) : le dispositif d'export automatique des bases à propos de la table de gestion « ne_pas_sauvegarder »)

Récupérer tout le contenu de /home/backup/postgresql en vérifiant la date des fichiers.

(D) 4.2 - Seconde étape : restauration

Après l'installation du nouveau serveur, l'enregistrement zephir, l'instanciation avec la bonne variante et toutes les mises à jour, copier depuis le serveur à migrer les fichiers récupérés à la première étape vers le répertoire /home/backup/postgresql/ du serveur Eole 2.8.x

Le fichier dumpall.dmp contient les commandes SQL nécessaires à la restauration des rôles. Ensuite, connecté en root dans le répertoire /root, on lance la commande :

```
psql -U postgres -d adl -f /home/backup/postgresql/dumpall.dmp
```

Pour le cas où cette restauration des rôles ait modifié le mot de passe du rôle « postgres », on redescend la configuration de PostgreSQL depuis le Zephir en lançant la commande

```
reconfigure
```

A ce stade, chaque base peut être restaurée indépendamment. Par exemple, pour la base toto dont le fichier de sauvegarde se nomme toto.backup et a été placé dans /home/backup/postgresql, utiliser la commande suivante depuis le répertoire /root :

```
pg_restore -U postgres -d adl -C -v /home/backup/postgresql/toto.backup
```

Cette commande crée automatiquement la base de données qui accueillera le contenu de la sauvegarde.

Si vous souhaitez restaurer une base de données issue d'une sauvegarde d'un serveur Microsoft, il est probable que l'encodage stipulé dans la sauvegarde ne soit pas reconnu par le serveur Eole qui est sous Ubuntu.

Après le *reconfigure* indiqué supra, il faut créer la base de données vide destinée à accueillir la sauvegarde.

```
psql -U postgres -c « create database toto template template_postgis ; »
```

```
pg_restore -d toto -v -U postgres /home/backup/postgresql/toto.backup
```

III. Point d'attention sur les messages de pg_restore

Après exécution de la commande `pg_restore`, l'outil affiche une information relative aux erreurs ignorées lors de la procédure de restauration.

```
pg_restore: warning: errors ignored on restore: 61
```

Dans l'exemple supra, nous avons restauré des données dans une base existante. Lors de l'exécution de la restauration, l'outil a tenté de créer les objets demandés dans le fichier backup alors que certains de ces objets existaient suite à la création préalable de la base.

Exemple : le fichier backup contient une commande de création de base DDT49 alors que celle-ci existe déjà.

```
pg_restore: error: could not execute query: ERROR: database "DDT49" already exists
```

Seule la lecture des messages fournis en mode verbeux (option `-v`) de `pg_restore` permet d'identifier si les erreurs décomptées relèvent d'une simple information ou d'une anomalie rencontrée lors de la procédure.



**MINISTÈRE
DE LA TRANSITION
ÉCOLOGIQUE
ET DE LA COHÉSION
DES TERRITOIRES**

*Liberté
Égalité
Fraternité*