

ESBL Géomatique V2.7

Migration des données





MINISTÈRE
DE LA TRANSITION
ÉCOLOGIQUE ET SOLIDAIRE
www.ecologique-solidaire.gouv.fr

MINISTÈRE
DE LA COHÉSION
DES TERRITOIRES
www.cohesion-territoires.gouv.fr

Historique des versions du document

Affaire suivie par

Christophe BOCQUET - CP2I/DOO
<i>Tél. : 02 40 12 84 07 / Fax : 02 40 12 84 44</i>
<i>Courriel :</i>

Rédacteur

Christophe BOCQUET- CP2I/DO Ouest

Relecteur

Prénom NOM - Service

Référence(s) intranet

<http://portail-ig.metier.i2>

SOMMAIRE

1 - INTRODUCTION.....	4
2 - DIFFÉRENTES STRATÉGIES POSSIBLES.....	4
3 - MÉTHODE 1 : RESTAURATION COMPLÈTE.....	4
3.1 - Première étape : sauvegarde.....	4
3.2 - Seconde étape : restauration.....	5
4 - MÉTHODE 2 : RESTAURATION BASE PAR BASE.....	6
4.1 - Première étape : sauvegarde.....	6
4.2 - Seconde étape : restauration.....	6

1 - Introduction

AVERTISSEMENT : Le présent document traite uniquement de la migration des bases de données entre un serveur ESBL V2.6.x et un serveur ESBL V2.7.x en variante géomatique. Ce document ne concerne **pas** les serveurs geobase.

Il n'est pas prévu de migration massive et obligatoire des serveurs. C'est à chaque service de choisir le moment opportun pour migrer son ou ses serveurs puisque la version 2.6 est encore maintenue jusqu'en juin 2021.

Il est cependant recommandé de ne pas attendre l'extrême limite pour migrer mais de planifier la migration à un moment où une interruption de service pourra être supportée.

2 - Différentes stratégies possibles

Il existe différentes méthodes et différents cas de figure peuvent se présenter dans les services. Volontairement, ce document se limite à deux cas de figure :

1 Installation de la version 2.7 sur la même machine, à la place de l'ancien système

Cette stratégie induit nécessairement une interruption de service mais ne nécessite pas un second serveur disponible.

2 Installation de la version 2.7 sur une autre machine, à côté du serveur en version 2.6

Cette stratégie 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.

Deux méthodes différentes sont décrites et elles peuvent être utilisées dans les deux cas.

3 - 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.

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 et de les récupérer 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 -file=/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.

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.

4 - 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
- ...

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.

Ensuite, récupérer tout le contenu de `/home/backup/postgresql` en vérifiant la date des fichiers.

4.2 - Seconde étape : restauration

Dans ce cas, il faut d'abord restaurer les rôles autres que postgres.

Copier les fichiers récupérés 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/`

Le fichier `dumpall.dmp` contient les commandes SQL nécessaires à la restauration des rôles. Il est à modifier en **supprimant** les deux lignes correspondant au rôle postgres au moyen d'un éditeur de texte (nano par exemple) :

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

Ensuite, connecté en root dans le répertoire `/root`, on lance la commande :

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

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 -C /home/backup/postgresql/toto.backup | psql -U postgres -d adl
```


