

## w.c.s. - Development #17726

### Ajouter un dump postgresql à la commande backup

20 juillet 2017 13:13 - Jean-Baptiste Jaillot

<b>Statut:</b>	Rejeté	<b>Début:</b>	20 juillet 2017
<b>Priorité:</b>	Normal	<b>Echéance:</b>	
<b>Assigné à:</b>	Benjamin Dauvergne	<b>% réalisé:</b>	0%
<b>Catégorie:</b>		<b>Temps estimé:</b>	0:00 heure
<b>Version cible:</b>		<b>Planning:</b>	
<b>Patch proposed:</b>	Non		
<b>Description</b>			
pg_dump -Oc de la base si mode postgresql activé.			

#### Historique

##### #1 - 20 juillet 2017 13:15 - Frédéric Péters

(Cette fois vraiment dans w.c.s.)

Il y a déjà du code différent (dans hobo.contrib.ozwillo) pour appeler le delete\_tenant de w.c.s.; plutôt qu'ajouter du code ici, j'appellerais la commande backup qui existe déjà pour produire un .tar.gz, qu'il est évidemment possible d'étendre pour inclure également un dump sql dedans.

Mais je ne mêlerais pas archivage et suppression dans la même commande.

##### #2 - 20 juillet 2017 13:17 - Jean-Baptiste Jaillot

Pour reprendre ce que disait Fred (et vérifié que j'ai bien compris): améliorer la commande backup pour ajouter le dump de la base (actuellement on ne dump que les fichiers du tenant).

##### #4 - 20 juillet 2017 13:18 - Jean-Baptiste Jaillot

- Assigné à mis à Jean-Baptiste Jaillot

##### #5 - 20 juillet 2017 14:44 - Benjamin Dauvergne

- Sujet changé de Améliorer la commande delete\_tenant à Ajouter un dump postgresql à la commande backup

- Description mis à jour

##### #6 - 20 juillet 2017 18:31 - Jean-Baptiste Jaillot

- Tracker changé de Bug à Development

##### #7 - 25 juillet 2017 01:36 - Jean-Baptiste Jaillot

- Fichier 0001-ctl-add-dump-in-backup-command-if-postgresql-is-true.patch ajouté

- Statut changé de Nouveau à En cours

- Patch proposed changé de Non à Oui

Hop. J'ai essayé de faire les tests mais je n'ai pas su faire avec la commande (et il n'y avait pas de test précédemment pour backup). Comme tout le code est dans la méthode execute, contrairement aux tests que j'ai pu voir sur les autres commandes (si c'est faisable je veux bien une astuce). Ou alors il faut déplacer le code dans une méthode qui est appelée dans execute (comme delete\_tenant par exemple).

Sinon j'ai testé sur la dev de sictiam et la preprod (pour avoir connexion locale et distante) sur l'instance sospel et tout marche bien. J'ai laissé le fonctionnement sur nom de fichier d'archive en option et je me base là dessus pour faire le dump et le zip (je récupère le dossier parent).

##### #8 - 25 juillet 2017 11:22 - Benjamin Dauvergne

Jean-Baptiste Jaillot a écrit :

Hop. J'ai essayé de faire les tests mais je n'ai pas su faire avec la commande (et il n'y avait pas de test précédemment pour backup). Comme tout le code est dans la méthode execute, contrairement aux tests que j'ai pu voir sur les autres commandes (si c'est faisable je veux bien une astuce). Ou alors il faut déplacer le code dans une méthode qui est appelée dans execute (comme delete\_tenant par exemple).

Comme tu as déjà trouvé la solution, je te propose de l'auto-accepter et de la développer.

« C'est curieux, chez les marins, ce besoin de faire des phrases ... »

Sinon j'ai testé sur la dev de sictiam et la preprod (pour avoir connexion locale et distante) sur l'instance sospel et tout marche bien. J'ai laissé le fonctionnement sur nom de fichier d'archive en option et je me base là dessus pour faire le dump et le zip (je récupère le dossier parent).

Bon c'est bizarre, dans un cas ça fait un tarball que ça pose soit dans `sub_options.filename`, soit dans `<tenant>/backups/<hostname>-backup-<date + ou - ISO zarbi>.tar.gz` et dans l'autre cas ça crée un zip dans `basedir(sub_options.filename)/<hostname>-<backup>-<date + ou - ISO zarbi>.zip` ou dans `<tenant>/backups/<idem>.zip`.

À moins qu'on soit extrêmement attaché à l'ancien fonctionnement j'aimerais bien voir une convergence de comportement avec la commande backup de hobo:

- avoir un paramètre `--destination` et pas `--filename`, toujours créer un zip, nommé le zip "`<application>__<hostname>__<date iso>.zip`"
- utiliser l'extension `.sql` pour le dump postgresql
- structurer tout ça,

en gros les mêmes remarques que sur [#17725](#)

#### #9 - 25 juillet 2017 11:31 - Frédéric Péters

Si on éloigne backup du comportement actuel, il faut faire suivre la commande restore.

#### #10 - 25 juillet 2017 11:42 - Benjamin Dauvergne

Frédéric Péters a écrit :

Si on éloigne backup du comportement actuel, il faut faire suivre la commande restore.

Ça me paraît un bon exercice.

#### #11 - 25 juillet 2017 18:39 - Jean-Baptiste Jaillet

Hum ok j'ai tout refait pour wcs, et je regarde restore. On veut juste changer les options, ou du coup avoir un restore qui peut récupérer un zip et reconstruire la base etc ? (ou il faut que la base existe ?).

Si oui il me semble mieux de faire un autre ticket que je puisse valider la partie backup et je fais le patch pour restore dans l'autre ticket. (dès que j'ai fini les tests je mets le patch, dans la soirée).

#### #12 - 25 juillet 2017 18:51 - Jean-Baptiste Jaillet

Je me suis auto répondu pour restore : <https://dev.entrouvert.org/issues/17790>. J'ai mis quelques questions de départ et suis preneur de conseils pour le départ.

#### #13 - 26 juillet 2017 02:11 - Jean-Baptiste Jaillet

- Fichier `0001-ctl-add-dump-in-backup-command-if-postgresql-is-true.patch` ajouté

Hop avec les tests pour backup (avec et sans sql) et restructuration du code.

J'ai fait une exception pour la commande que je raise en cas d'erreur, mais je sais pas si on préfère faire un `print >> stderr` et un `return 1` pour les commandes de wcs.

```
try:
    self.backup_instance(pub, sub_options, args)
except CmdError as e:
    print >> sys.stderr, e
    return 1
```

Un truc dans le style (dans execute).

#### #14 - 26 juillet 2017 09:03 - Frédéric Péters

```
os.mkdir('/tmp/test')
```

Tu ne peux pas toucher comme ça aux /tmp des gens. `tempfile.mkdtemp` fait le job correct manuellement mais tu peux aussi regarder à <https://docs.pytest.org/en/latest/tmpdir.html#the-tmpdir-factory-fixture> pour quelque chose intégré à pytest.

```
if not any(z for z in os.listdir('/tmp/test') if '.zip' in z):
    assert False
```

Idiomatiquement, on ferait plutôt, en Python et particulièrement dans w.c.s. : `assert [x for x in os.listdir('/tmp/test') if x.endswith('.zip')]`

`datetime.datetime.now().date()` c'est `datetime.date.today()`.

Pas fan du tar.gz dans un zip dans un cas et pas l'autre; et comme pas fan de l'idée d'un tar.gz dans un zip, pourquoi ne pas tout taper, répertoires de w.c.s. et dump de la db dans le même fichier (peu importe que ça soit zip ou tar.gz).

```
subprocess.check_call(['pg_dump', '-Oc', '-h', db_host,  
                      '-U', db_user, '-p', db_port,  
                      '-w', '-f', dump_path, '-d', db_name])
```

Dans la commande qui supprime un tenant de w.c.s., on peut se trouver à créer un schéma dans la db; si jamais ensuite la db est réutilisée, le dump ici il va également prendre ces "vieilles" infos, je préciserais --schema=public.

#### #15 - 26 juillet 2017 09:22 - Benjamin Dauvergne

Frédéric Péters a écrit :

Pas fan du tar.gz dans un zip dans un cas et pas l'autre; et comme pas fan de l'idée d'un tar.gz dans un zip, pourquoi ne pas tout taper, répertoires de w.c.s. et dump de la db dans le même fichier (peu importe que ça soit zip ou tar.gz).

L'avantage que je voyais au tar.gz c'est le support des liens symboliques qui sont notamment utilisés par storage.py, mais qu'on pourrait avoir autrement aussi. Alors peut-être qu'on devrait plutôt faire un tar.gz tout court et zipper !

#### #16 - 26 juillet 2017 16:12 - Jean-Baptiste Jaillet

- Fichier 0001-ctl-add-dump-in-backup-command-if-postgresql-is-true.patch ajouté

Ok, j'ai changé tout ça. Je suis parti du principe qu'on met tout dans l'archive tar.gz du coup (en gros la même archive, et quand y'a postgresql à True, y'a un dump en plus).

J'ai aussi implémenté ce que je disais plus haut sur la gestion de mes exception pour que la partie UI soit un peu mieux (ça collait la trace de l'exception sinon).

Les tests fonctionnent, testé sur sictiam.

#### #17 - 07 septembre 2017 19:58 - Benjamin Dauvergne

- Assigné à changé de Jean-Baptiste Jaillet à Benjamin Dauvergne

Je reprends ce ticket, je vais ajouter un test, Fred tu peux me confirmer que les fonctionnalités te vont bien comme ça ? Je vais renommer la commande sur hobo en backup au lieu de backup\_tenant. Il va rester une petite différence vu que coté w.c.s. on prend le hostname comme paramètre général et coté hobo via l'option --domain. À voir si on veut faire converger ça à un moment et dans quel sens.

#### #18 - 04 mars 2019 11:09 - Benjamin Dauvergne

- Statut changé de En cours à Nouveau

- Patch proposed changé de Oui à Non

#### #19 - 04 mars 2019 11:09 - Benjamin Dauvergne

- Assigné à Benjamin Dauvergne supprimé

#### #20 - 04 mars 2019 22:32 - Benjamin Dauvergne

- Assigné à mis à Benjamin Dauvergne

#### #21 - 05 mars 2019 08:40 - Frédéric Péters

Personne n'utilise ou n'utilisera cette commande, qu'il faudrait sinon porter en management command, je serais plutôt pour la supprimer.

#### #22 - 05 mars 2019 10:18 - Benjamin Dauvergne

De fait c'est utilisé au sictiam pour le deprovisionning (dans sa version actuelle sans backup SQL) donc la question est plutôt est-ce qu'on veut développer un vrai deprovisionning dans hobo, ou plus vraisemblable on statue que ça n'arrive qu'un fois par décade et donc c'est très bien à la main.

#### #23 - 05 mars 2019 10:19 - Benjamin Dauvergne

- Statut changé de Nouveau à Rejeté

### Fichiers

0001-ctl-add-dump-in-backup-command-if-postgresql-is-true.patch	3,08 ko	24 juillet 2017	Jean-Baptiste Jaillet
0001-ctl-add-dump-in-backup-command-if-postgresql-is-true.patch	9,18 ko	26 juillet 2017	Jean-Baptiste Jaillet
0001-ctl-add-dump-in-backup-command-if-postgresql-is-true.patch	8,35 ko	26 juillet 2017	Jean-Baptiste Jaillet