

BiJoe - Bug #41030

ProgrammingError: relation "dates" does not exist

26 mars 2020 09:24 - Sentry Io

Statut:	Nouveau	Début:	26 mars 2020
Priorité:	Normal	Echéance:	
Assigné à:		% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	Non
Patch proposé:	Non		
Description			
https://sentry.entrouvert.org/entrouvert/publik/issues/597/			
ProgrammingError: relation "dates" does not exist LINE 1: ...data.id) AS count FROM formdata RIGHT OUTER JOIN dates AS "... ^			
(6 additional frame(s) were not displayed)			
... File "rest_framework/views.py", line 466, in dispatch response = self.handle_exception(exc) File "rest_framework/views.py", line 463, in dispatch response = handler(request, *args, **kwargs) File "bijoe/visualization/views.py", line 340, in get data = cell_value(visualization.data()[0].measures[0]) File "bijoe/visualization/utils.py", line 162, in data [self.measure])) File "bijoe/engine.py", line 437, in query cursor.execute(sql)			
Internal Server Error: /visualization/33/json/			

Historique

#1 - 26 mars 2020 09:24 - Lauréline Guérin

- Projet changé de Suivi des traces à BiJoe

#2 - 26 mars 2020 10:36 - Benjamin Dauvergne

La mise à jour de la table date n'est pas transactionnelle, pendant le rechargement des données par wcs-olap la table est supprimée puis recrée, donc pendant un temps non nul elle n'est plus là ./

```
def do_dates_table(self):
    self.ex("DROP TABLE IF EXISTS public.dates")
    last_date = datetime.datetime.today().replace(month=12, day=31)
    self.ex('''
CREATE TABLE public.dates AS (SELECT
    the_date.the_date::date AS date,
    to_char(the_date.the_date, 'TMday') AS day,
    to_char(the_date.the_date, 'TMmonth') AS month
FROM
    generate_series('2010-01-01'::date, '%s'::date, '1 day'::interval)
AS the_date(the_date));''' % last_date.strftime('%Y-%m-%d'))
```

Le plus simple ce serait de ne plus la créer dans le schéma public, sinon plus propre ajouter simplement les dates manquantes.

#3 - 26 mars 2020 10:40 - Benjamin Dauvergne

Donc si quelqu'un veut s'en occuper :

Solution 1 : remplacer public.dates par dates, ne plus ajouter public au search_path (ligne 99 de feeder.py)

Solution 2 : vérifier dans information_schema.tables si dates existe, si non créer, puis générer la série des dates manquantes et les ajouter INSERT .. FROM (SELECT ... FROM generate_series(SELECT MAX(date) + '1 day'::interval FROM dates, ...) ...