

## OLAP / Business Intelligence pour Publik - Bug #25982

### rendre les identifiants stables sinon les visualisation ne peuvent pointer sur des valeurs

31 août 2018 09:59 - Serghei Mihai

<b>Statut:</b>	Fermé	<b>Début:</b>	31 août 2018
<b>Priorité:</b>	Normal	<b>Echéance:</b>	
<b>Assigné à:</b>	Emmanuel Cazenave	<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>	Oui		
<b>Description</b>			
Car les identifiants, séquentiels, sont générés au fur et à mesure de la réception des données depuis wcs. Et donc l'ordre peut être différent.			
Lors du calcul des stats faire la résolution "label" -> "id".			
<b>Demandes liées:</b>			
Lié à OLAP / Business Intelligence pour Publik - Development #30752: Conserve...		<b>Fermé</b>	<b>19 février 2019</b>

#### Historique

##### #1 - 04 septembre 2018 11:04 - Benjamin Dauvergne

Oui, pour les ItemField (choix fermés) dont on a pas la visibilité de l'ensemble, en fait uniquement ceux avec une source JSONP, on accumule les valeurs au fur et à mesure du chargement et on génère un id numérique, l'ordre peut donc varier et donc les ids. Pour les types JSON, formule ou liste explicite c'est moins fréquent parce qu'on a l'ensemble des choix exportés dans le schéma du formulaire mais on aura aussi un problème en cas de changement de la liste.

La bonne solution comme le suggère subtilement Serghei ce serait de stocker le label en plus de l'id dans le stockage de la visualisation et chercher le label au dernier moment au lieu d'utiliser l'id si c'est possible (en cas de suppression d'une valeur on aura encore un souci mais là on y peut rien).

##### #2 - 04 septembre 2018 11:08 - Emmanuel Cazenave

- Assigné à mis à Emmanuel Cazenave

##### #3 - 04 septembre 2018 12:28 - Thomas Noël

Benjamin Dauvergne a écrit :

Pour les types JSON, formule ou liste explicite c'est moins fréquent parce qu'on a l'ensemble des choix exportés dans le schéma du formulaire mais on aura aussi un problème en cas de changement de la liste.

Et en réalité c'est assez fréquent car il faut compter sur les JSON qui sont dépendants d'une variable précédente dans le formulaire, genre « la liste des quartiers de la ville {{form\_var\_ville}} ». Je dis ça juste pour faire mon intéressant dans le ticket, la solution subtile de Serghei est la bonne.

##### #5 - 07 septembre 2018 14:10 - Emmanuel Cazenave

- Statut changé de Nouveau à En cours

##### #6 - 14 septembre 2018 10:20 - Emmanuel Cazenave

- Statut changé de En cours à Nouveau

- Assigné à Emmanuel Cazenave supprimé

##### #8 - 19 septembre 2018 18:06 - Emmanuel Cazenave

- Assigné à mis à Emmanuel Cazenave

##### #9 - 18 octobre 2018 18:05 - Benjamin Dauvergne

- Fichier 0001-reuse-older-id-for-open-item-fields-fixes-25982.patch ajouté

- Statut changé de Nouveau à Solution proposée

- Patch proposed changé de Non à Oui

### #10 - 18 octobre 2018 18:12 - Benjamin Dauvergne

- Statut changé de Solution proposée à En cours

Ne marche pas si les valeurs n'arrivent pas dans le bon ordre<sup>1</sup>, il faut plutôt récupérer totalement l'ancienne lors de la création de la table.

<sup>1</sup> Ancienne table

id	value
1	a
2	b
3	c

Nouvelles valeurs dans leur ordre d'arrivée: a, c, d, c, donnant la table:

id	value
1	a
2	b
3	c

ici erreur lors de l'insertion de (3, c) repris de l'ancienne table.

### #11 - 18 octobre 2018 18:18 - Benjamin Dauvergne

- Fichier 0001-reuse-older-id-for-open-item-fields-fixes-25982.patch ajouté

- Statut changé de En cours à Solution proposée

Nouvelle solution, a la creation d'une table pour un "open item field" on importe la table du précédent schéma (dont les valeurs ne disparaîtront donc plus).

### #12 - 23 octobre 2018 15:54 - Benjamin Dauvergne

- Sujet changé de dans les visualisations utiliser les libellés des champs filtres au lieu des identifiants à rendre les identifiants stables sinon les visualisation ne peuvent pointer sur des valeurs

### #13 - 23 octobre 2018 15:55 - Benjamin Dauvergne

- Projet changé de BiJoe à OLAP / Business Intelligence pour Publik

Le problème semble aussi présent sur les tables de formulaires voir #27529.

### #15 - 25 octobre 2018 16:32 - Benjamin Dauvergne

- Statut changé de Solution proposée à En cours

### #16 - 26 octobre 2018 11:13 - Benjamin Dauvergne

Bouh c'est plus compliqué que prévu, garder des valeurs de formdef pour des formdef qui n'existent plus et dont la table aura disparu c'est embêtant (jusqu'à ce qu'on sache gérer de l'incrémental).

Il faudrait:

- pour les formdef et les agents utiliser un identifiant stable (on a id pour les agents, pour les formdefs id+slug je dirait, le premier pouvant être réutiliser il me semble, mais le deuxième pouvant changer :/)
- pour les valeurs des options peut-être garder le plan initiale de conserver dans la visualisation le label et non la valeur...

### #17 - 09 novembre 2018 18:12 - Serghei Mihai

Je serai pour ici gérer uniquement la cohérence des libellés et valeurs dans les visualisations.

Le cas des formdefs inexistantes peut-être géré séparément, dans [#27412](#) par exemple.

### #18 - 09 janvier 2019 12:03 - Serghei Mihai

1. take old id, value paris an inject them <--- s/paris/paris/

et

```
selx.ex('SELECT SETVAL...
```

il faut un self.ex

#### #19 - 09 janvier 2019 15:28 - Benjamin Dauvergne

Non mais de toute façon je ne suis pas content de cette solution... Le seul chemin viable pour moi c'est de modifier complètement les identifiants mais je ne vois pas comment en même temps modifier les visualisations parce que la modification sera faite dans wcs-olap et pas dans bijoe et je ne vois pas comment faire une migration dans bijoe pour cela...

Un plan en plusieurs temps :

1. ajouter les identifiants naturels au coté wcs-olap sans rien changer d'autres
2. quand tout ça est à jour, passer une migration manuelle sur les instances bijoe qui viendra modifier les visualisations pour utiliser cette identifiant plutôt que la clé primaire de postgres
3. modifier le schéma généré par wcs-olap pour utiliser cette valeur la jointure plutôt que la clé primaire

#### #20 - 18 janvier 2019 15:30 - Serghei Mihai

Du coup on fait des tickets séparés: une créer des identifiants naturels, et un autre pour faire la migration et modifier les schéma?

#### #21 - 18 janvier 2019 15:44 - Benjamin Dauvergne

Serghei Mihai a écrit :

Du coup on fait des tickets séparés: une créer des identifiants naturels, et un autre pour faire la migration et modifier les schéma?

On peut faire le point 1 dans ce ticket:

- identifiants stables pour les agents
- les rôles
- les fonctions
- les statuts
- les items (soit l'id pour une source de donnée, soit le libellé)

On aura des tables id,source\_id,label et pour l'instant source\_id sera inutilisé par BiJoe, sans modifier le schéma il continuera à utiliser id, pour pas se faire chier je propose que source\_id soit toujours varchar et quand on a un identifiant numérique on fait juste appel à str().

#### #22 - 18 janvier 2019 15:51 - Benjamin Dauvergne

- Assigné à Emmanuel Cazenave supprimé

J'enlève Manu comme assigné, je ne sais pas qui va s'en occuper.

#### #23 - 11 février 2019 15:22 - Emmanuel Cazenave

- Assigné à mis à Emmanuel Cazenave

#### #24 - 11 février 2019 15:44 - Benjamin Dauvergne

Discussion à l'instant avec manu, plan en 2,5 temps :

- pour chaque table de label, refaire la mécanique pour prendre la table dans l'ancien schéma, insérer la totalité des valeurs dans la nouvelle table et poser le générateur de séquence comme le max des id existants (pour les nouvelles valeurs), voir si on préfère conserver la totalité des anciennes valeurs ou ne garder que celle existant vraiment dans le nouveau schéma
- préparer l'avenir en ajoutant aux tables de label une colonne natural\_id varchar dans laquelle on mettra :
  - pour les formdef, formdef.id
  - pour les category, category.id
  - pour les items field "naturels", le label de l'item
  - pour les items field venant de source de données form\_var\_<varname>\_raw (l'id naturel venant de la source de donnée)
  - pour les agents le NameID
  - pour les rôles, le slug, si pas de slug on ignore (ça ne doit pas arriver)
  - pour les canaux, le slug (le label en fait) : web, mail, etc..
  - pour les statuts simplifiés, rien, id totalement contrôlé par wcs-olap pas de souci

À la fin ouvrir un nouveau ticket pour la migration conjointe entre wcs-olap et bijoe des ids générés vers les natural\_id:

- coté wcs-olap changer tous les 'type': 'integer', 'value': "field\_x.id" en 'type': 'string', 'value': "field\_x.natural\_id"
- coté bijoe dans toutes les visualisations chercher les filtrages dans toutes les visualisation et quand on trouve une valeur entière la remplacer par le natural\_id qui correspond en regardant dans la table concernée:
  - for x in filters: puis trouver la table formdata\_y\_field\_x correspondante, cherche l'id dans la colonne id et recopier le natural\_id

## #25 - 11 février 2019 16:01 - Serghei Mihai

Benjamin Dauvergne a écrit :

Discussion à l'instant avec manu, plan en 2,5 temps :

- pour chaque table de label, refaire la mécanique pour prendre la table dans l'ancien schéma, insérer la totalité des valeurs dans la nouvelle table et poser le générateur de séquence comme le max des id existants (pour les nouvelles valeurs), voir si on préfère conserver la totalité des anciennes valeurs ou ne garder que celle existant vraiment dans le nouveau schéma

Je suis d'avis de garder toutes les valeurs même si elles n'existent pas. Et cela pour ne pas casser les visualisations sauvegardées avec ces libellés.

## #26 - 11 février 2019 17:31 - Benjamin Dauvergne

Serghei Mihai a écrit :

Benjamin Dauvergne a écrit :

Discussion à l'instant avec manu, plan en 2,5 temps :

- pour chaque table de label, refaire la mécanique pour prendre la table dans l'ancien schéma, insérer la totalité des valeurs dans la nouvelle table et poser le générateur de séquence comme le max des id existants (pour les nouvelles valeurs), voir si on préfère conserver la totalité des anciennes valeurs ou ne garder que celle existant vraiment dans le nouveau schéma

Je suis d'avis de garder toutes les valeurs même si elles n'existent pas. Et cela pour ne pas casser les visualisations sauvegardées avec ces libellés.

Mouais, faudra voir, peut-être ajouter une colonne alive booléenne pour noter si la valeur vient d'avant ou est encore utilisée, juste pour se souvenir qu'on y pense.

## #27 - 19 février 2019 11:23 - Emmanuel Cazenave

- Lié à *Development* #30752: *Conserver les données des tables de dimensions ajouté*

## #28 - 27 mars 2019 15:41 - Emmanuel Cazenave

- Statut changé de *En cours* à *Fermé*

Les choses se sont passées ici : [#30752](#), la suite se passera là [#31315](#).

## Fichiers

---

0001-reuse-older-id-for-open-item-fields-fixes-25982.patch	1,89 ko	18 octobre 2018	Benjamin Dauvergne
0001-reuse-older-id-for-open-item-fields-fixes-25982.patch	1,39 ko	18 octobre 2018	Benjamin Dauvergne