

## w.c.s. - Bug #54889

### sources de données agendas automatiques, changement de nom et incohérence lors de l'export/import

15 juin 2021 14:40 - Thomas Noël

<b>Statut:</b>	Fermé	<b>Début:</b>	15 juin 2021
<b>Priorité:</b>	Normal	<b>Echéance:</b>	
<b>Assigné à:</b>	Lauréline Guérin	<b>% réalisé:</b>	0%
<b>Catégorie:</b>		<b>Temps estimé:</b>	0:00 heure
<b>Version cible:</b>		<b>Planning:</b>	Non
<b>Patch proposed:</b>	Oui		

**Description**  
(d'un client perspicace)

Le changement de nom d'un agenda après création des sources de données automatiques associées, ne se répercute pas sur l'identifiant de la source de données (même après actualisation de l'agenda).

Dans le principe ça semble plutôt une bonne chose, puisque l'ancienne source de données est possiblement utilisée dans le formulaire et continue donc de fonctionner...

En revanche, c'est problématique lors de l'export/import : l'import de l'agenda donne lieu à la création d'une source de donnée dont l'identifiant est nouveau (ne se base pas sur celui exporté) : l'import d'un formulaire par la suite est donc naturellement "bloqué" car la source de donnée est inconnue.

#### Révisions associées

##### Révision f48199fa - 25 juin 2021 10:23 - Lauréline Guérin

datasource: set slug of chrono datasources (#54889)

#### Historique

##### #1 - 15 juin 2021 14:41 - Thomas Noël

- *Projet changé de Publik à Chrono*

- *Club Non supprimé*

Je pose ça dans Chrono car une piste pourrait être que le système d'import d'agendas propose/impose le respect des identifiants exportés.

##### #3 - 15 juin 2021 14:43 - Frédéric Péters

Tu parles d'identifiant ("identifiant est nouveau") les sources de données ne se basent pas sur les slugs ? (je vérifie et elles le font).

Le soucis ici c'est qu'un export/import ne conserverait pas les slugs ?

##### #4 - 15 juin 2021 14:45 - Thomas Noël

Je ne sais pas trop de quel identifiant on parle finalement, car dans l'import\_json d'un agenda je vois bien :

```
agenda, created = cls.objects.get_or_create(slug=data['slug'], defaults=data)
```

donc a priori effectivement les slugs ne devraient pas changer.

Je vais demander des détails au client.

##### #5 - 15 juin 2021 16:10 - Lauréline Guérin

dans wcs:

```
datasource = existing_datasources.get(url)
if datasource is None:
    datasource = NamedDataSource()
    datasource.external = 'agenda'
    datasource.data_source = {'type': 'json', 'value': url}
datasource.external_status = None # reset
```

```
datasource.name = agenda['text']
datasource.store()
```

On ne précise pas le slug de la datasource, il est généré à partir du name.  
Si l'agenda change de label, agenda['text'] aussi, on se retrouve avec une nouvelle source de donnée.

=> il faudrait prendre agenda['id'] (== slug de l'agenda côté chrono) pour en faire un slug de datasource ?

note: comment ne pas casser les sources de données agenda existantes lors d'un refresh, si on change ça ?

#### #6 - 15 juin 2021 17:09 - Thomas Noël

- *Projet changé de Chrono à w.c.s.*

Lauréline Guérin a écrit :

=> il faudrait prendre agenda['id'] (== slug de l'agenda côté chrono) pour en faire un slug de datasource ?

Ouaip.

note: comment ne pas casser les sources de données agenda existantes lors d'un refresh, si on change ça ?

Je crois qu'on pourrait chercher à avoir un fonctionnement à la "update\_or\_create" :

- si une datasource de type external=agenda avec une même id existe déjà, alors on la met à jour
- sinon on fait comme le code actuelle, juste un store() qui va se débrouiller à créer un nouvel id

(et je passe ce ticket sur w.c.s. parce que c'est bien là qu'est le "bogue", donc)

#### #7 - 18 juin 2021 10:39 - Lauréline Guérin

- *Projet changé de w.c.s. à Chrono*

- *Assigné à mis à Lauréline Guérin*

#### #8 - 18 juin 2021 10:40 - Lauréline Guérin

- *Projet changé de Chrono à w.c.s.*

#### #9 - 18 juin 2021 11:40 - Lauréline Guérin

- *Fichier 0001-datasource-set-slug-of-chrono-datasources-54889.patch ajouté*

- *Statut changé de Nouveau à Solution proposée*

- *Patch proposed changé de Non à Oui*

voilà, ça ne touche pas à l'existant; seules les nouvelles datasources prendront ce nouveau slug

#### #10 - 18 juin 2021 11:57 - Thomas Noël

« datasource3.slug == 'chrono-ds-slug-A' » il ne faut pas avoir de "-" sinon on ne peut pas utiliser ces slug dans des expressions du genre data\_source.agenda\_truc (rarement utilisé, mais.). Ajoute une passe de "misc.simplify(..., space='\_')" si tu veux t'assurer le truc.

(Je constate d'ailleurs un bogue dans get\_new\_slug qui fait des xxx-1 / xxx-2 quand le slug existe déjà, zou, [#54979](#))

Aussi, quand un slug "chrono\_ds\_slug\_A" est déjà pris on pourrait chercher à avoir "chrono\_ds\_slug\_A\_1" et donc faire que get\_new\_slug puisse partir d'une base « def get\_new\_slug(self, base=None) » -- tu pourrais alors t'éviter le try/except dans la création d'une nouvelle source et faire juste « datasource.slug = datasource.get\_new\_slug(base='chrono\_ds\_%s' % agenda['slug']) »

#### #11 - 18 juin 2021 14:46 - Lauréline Guérin

- *Fichier 0001-datasource-set-slug-of-chrono-datasources-54889.patch ajouté*

#### #12 - 25 juin 2021 09:51 - Frédéric Péters

- *Statut changé de Solution proposée à Solution validée*

#### #13 - 25 juin 2021 11:02 - Lauréline Guérin

- *Statut changé de Solution validée à Résolu (à déployer)*

commit f48199fa92549d89c9c9cbcef19aa97370032

Author: Lauréline Guérin <zebuline@entrouvert.com>

Date: Fri Jun 18 11:38:50 2021 +0200

datasource: set slug of chrono datasources (#54889)

#### #14 - 29 juin 2021 17:16 - Frédéric Péters

- Statut changé de Résolu (à déployer) à Solution déployée

#### Fichiers

---

0001-datasource-set-slug-of-chrono-datasources-54889.patch	8,63 ko	18 juin 2021	Lauréline Guérin
0001-datasource-set-slug-of-chrono-datasources-54889.patch	9,83 ko	18 juin 2021	Lauréline Guérin