

Chrono - Development #47916

Incohérences lors de la duplication d'un guichet avec des sources d'exceptions

21 octobre 2020 11:19 - Valentin Deniaud

Statut:	Fermé	Début:	21 octobre 2020
Priorité:	Normal	Echéance:	
Assigné à:	Valentin Deniaud	% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	Non
Patch proposed:	Oui		
Description			
Sens, chaque exception est en double et la source est en double : https://agendas.demarches.ville-sens.fr/manage/agendas/56/settings			
Essonne, chaque exception est en double, une seule source, le doublon a perdu son attribut « source » et son flag « external » : https://agendas.demarches.essonne.fr/manage/agendas/1/settings			
Je regarde...			

Révisions associées

Révision f880490e - 22 octobre 2020 15:24 - Valentin Deniaud

agendas: set proper exception source when duplicating (#47916)

Révision bb3e011b - 22 octobre 2020 15:24 - Valentin Deniaud

agendas: do not import exception from settings when duplicating (#47916)

Révision c1d5eb32 - 22 octobre 2020 16:45 - Valentin Deniaud

agendas: add unicity constraint on exception source slug (#47916)

Historique

#1 - 21 octobre 2020 11:47 - Valentin Deniaud

Et donc le coupable, « Copier le paramétrage du guichet »...

En plus du patch pour éviter cette duplication, il va falloir une migration qui ajoute une contrainte d'unicité sur le slug de la source d'exception, et vire les doublons au préalable.

#2 - 21 octobre 2020 11:59 - Valentin Deniaud

Side effect, ça fait des exceptions qu'on ne peut plus désactiver, #47915.

#3 - 21 octobre 2020 18:08 - Valentin Deniaud

- Fichier 0001-agendas-fix-duplication-of-desk-with-external-except.patch ajouté
- Fichier 0002-agendas-add-unicity-constraint-on-exception-source-s.patch ajouté
- Tracker changé de Bug à Development
- Statut changé de Nouveau à Solution proposée
- Patch proposed changé de Non à Oui

0001 pour fixer le bug, 0002 pour réparer les db et rajouter une contrainte pour être sûr que ça n'arrive plus...

La migration de 0002 doit se faire en deux fois, parce que une migration = un bloc atomic, et que django ne peut pas bouger les données et ajouter des contraintes au champs en même temps (erreur « pending trigger events »). En vrai c'est pas fou, on aimerait bien de l'atomicité, je m'y repenche demain.

#4 - 22 octobre 2020 10:40 - Valentin Deniaud

Valentin Deniaud a écrit :

je m'y repenche demain.

C'est un comportement documenté et je n'ai pas l'impression qu'il y ait de contournement, <https://docs.djangoproject.com/en/1.11/ref/migration-operations/#django.db.migrations.operations.RunPython> :

you should avoid combining schema changes and RunPython operations in the same migration or you may hit errors like OperationalError: cannot ALTER TABLE "mytable" because it has pending trigger events

Mais en fait c'est un faux problème que je me posais : comme la première migration assure l'unicité, et que la deuxième ajoute la contrainte, je me disais qu'un doublon pouvait très hypothétiquement être créé dans l'intervalle et faire échouer le truc. Mais un doublon ne peut plus être créé, c'est assuré par le premier patch. Donc tout va bien.

#5 - 22 octobre 2020 11:52 - Valentin Deniaud

- Statut changé de Solution proposée à En cours

Info pour qui relira, tapé sur le salon, à propos de 0001 :
Il y a deux trucs à corriger

- ne pas importer les exceptions si on est dans un cas de duplication (et laisser ensuite la duplication de source se faire, on pourrait faire le choix inverse mais ça serait moins clair)
- ne pas dupliquer les exceptions externes, parce qu'elles sont liées à la source du guichet initial et on veut pas ça

Et en tapant ça je m'aperçois que le deuxième point n'est pas spécifique aux exceptions jours fériés, c'est valable avec toutes les sources (import d'un ICS -> duplication -> suppression de l'ICS dans le guichet dupliqué (rien ne se passe) -> suppression de l'ICS dans le fichier initial (exception supprimées dans les deux guichets).

#8 - 22 octobre 2020 12:46 - Valentin Deniaud

- Fichier 0002-agendas-do-not-import-exception-from-settings-when-d.patch ajouté
- Fichier 0003-agendas-add-unicity-constraint-on-exception-source-s.patch ajouté
- Fichier 0001-agendas-set-proper-exception-source-when-duplicating.patch ajouté
- Statut changé de En cours à Solution proposée

Voilà, en différenciant les deux points du dessus dans les deux premiers patches.

#9 - 22 octobre 2020 15:46 - Valentin Deniaud

Valentin Deniaud a écrit :

Et en tapant ça je m'aperçois que le deuxième point n'est pas spécifique aux exceptions jours fériés, c'est valable avec toutes les sources (import d'un ICS -> duplication -> suppression de l'ICS dans le guichet dupliqué (rien ne se passe) -> suppression de l'ICS dans le fichier initial (exception supprimées dans les deux guichets).

Mais ça s'autocorrige tout seul au bout d'une heure avec la commande qui synchronise les ICS (cycle qui supprime/remet les exceptions), c'est pour ça que c'est passé sous les radars. Pas besoin de migration qui rattrape, donc.

#10 - 22 octobre 2020 16:03 - Valentin Deniaud

- Sujet changé de Source d'exception jours fériés, n'importe quoi sur la prod à Incohérences lors de la duplication d'un guichet avec des sources d'exceptions

#11 - 22 octobre 2020 16:07 - Benjamin Dauvergne

0001 : ok
0002 : ok
0003 : ok, juste je me demande si prendre le premier des doublons d'une source ayant des exceptions est suffisant, par rapport à leur état (source.enabled)

Note pour moi : pour le 0003 il faut juste se mettre dans l'idée quand on lit qu'en fait les sources passent leur temps à effacer toutes les exceptions qui leur sont liées à chaque fois qu'elles se mettent à jour, donc en fait on a très peu de chance de trouver des exceptions incohérentes (liée au mauvais bureau, j'ai eu un peu du mal à comprendre pourquoi on pouvait en faire aussi peu) et supprimer les sources en trop suffit (aucune source ne manque et il n'y a pas de doublon de source pour le type de source "ics")

On peut peut-être remplacer ça par un update_or_create() maintenant qu'on a un index d'unicité :

```
try:
    source = TimePeriodExceptionSource.objects.get(desk=self, settings_slug=slug)
except TimePeriodExceptionSource.DoesNotExist:
```

```
source = TimePeriodExceptionSource.objects.create(
    desk=self, settings_slug=slug, enabled=False
)
```

#12 - 22 octobre 2020 16:08 - Benjamin Dauvergne

- Statut changé de Solution proposée à Solution validée

#13 - 22 octobre 2020 16:33 - Valentin Deniaud

Thx pour la relecture.

Benjamin Dauvergne a écrit :

0003 : ok, juste je me demande si prendre le premier des doublons d'une source ayant des exceptions est suffisant, par rapport à leur état (source.enabled)

Quand tu dupliques une première fois, tu te retrouves avec deux sources enabled, l'une créée par l'import lors du save(), l'autre par la duplication. On veut garder la première, qui a correctement créé ses exceptions, et pas la seconde. Donc on voit qu'on ne peut pas se fier à enabled pour prendre des décisions ici.

Lors des duplications suivantes, on peut se retrouver avec plusieurs sources enabled qui ont des exceptions, il suffit d'en choisir une au pif.

On peut peut-être remplacer ça par un update_or_create() maintenant qu'on a un index d'unicité :

En vrai je relis cette ligne et je ne comprends pas pourquoi il n'y a pas un get_or_create (certes il n'y a pas d'index d'unicité mais je ne pense pas que ce soit la raison vu que le même pb se pose avec get tout court). Par contre update_or_create je pense pas, on ne veut pas mettre à jour la valeur de enabled. Trop frileux pour y toucher ici en tout cas.

#14 - 22 octobre 2020 16:40 - Valentin Deniaud

Valentin Deniaud a écrit :

Benjamin Dauvergne a écrit :

0003 : ok, juste je me demande si prendre le premier des doublons d'une source ayant des exceptions est suffisant, par rapport à leur état (source.enabled)

Quand tu dupliques une première fois, tu te retrouves avec deux sources enabled, l'une créée par l'import lors du save(), l'autre par la duplication. On veut garder la première, qui a correctement créé ses exceptions, et pas la seconde. Donc on voit qu'on ne peut pas se fier à enabled pour prendre des décisions ici.

Lors des duplications suivantes, on peut se retrouver avec plusieurs sources enabled qui ont des exceptions, il suffit d'en choisir une au pif.

Bon aller à la rigueur il y a un cas où on peut se permettre d'être plus intelligent :

- duplication comme je décris, le guichet dupliqué a une source enabled avec des exceptions, l'autre enabled sans
- quelqu'un vient désactiver la source avec des exceptions
- la migration risque de garder la source enabled qui n'a pas d'exceptions, alors qu'on aurait voulu celle dans un état cohérent.

J'ajoute un petit if qui gère ça et je pousse.

#15 - 22 octobre 2020 16:45 - Valentin Deniaud

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

```
commit c1d5eb32ee80623810c8a016d33e612bec0b1ffb
Author: Valentin Deniaud <vdeniaud@entrouvert.com>
Date: Wed Oct 21 17:47:38 2020 +0200
```

```
agendas: add unicity constraint on exception source slug (#47916)
```

```
commit bb3e011beaec6dddedc0a30698e03e74f8f95bb3
Author: Valentin Deniaud <vdeniaud@entrouvert.com>
Date: Thu Oct 22 11:59:31 2020 +0200
```

```
agendas: do not import exception from settings when duplicating (#47916)
```

```
commit f880490e00e82986733cbcf2369d6c5f7e80213f
Author: Valentin Deniaud <vdeniaud@entrouvert.com>
Date: Thu Oct 22 11:58:55 2020 +0200
```

```
agendas: set proper exception source when duplicating (#47916)
```

#16 - 22 octobre 2020 18:16 - Frédéric Péters

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

Fichiers

0001-agendas-fix-duplication-of-desk-with-external-except.patch	3,42 ko	21 octobre 2020	Valentin Deniaud
0002-agendas-add-unicity-constraint-on-exception-source-s.patch	8,05 ko	21 octobre 2020	Valentin Deniaud
0002-agendas-do-not-import-exception-from-settings-when-d.patch	2,87 ko	22 octobre 2020	Valentin Deniaud
0003-agendas-add-unicity-constraint-on-exception-source-s.patch	8,05 ko	22 octobre 2020	Valentin Deniaud
0001-agendas-set-proper-exception-source-when-duplicating.patch	3,31 ko	22 octobre 2020	Valentin Deniaud