

Chrono - Bug #27263

crash sur import d'un ics

11 octobre 2018 17:06 - Thomas Noël

Statut:	Fermé	Début:	11 octobre 2018
Priorité:	Normal	Echéance:	
Assigné à:	Benjamin Dauvergne	% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	
Patch proposé:	Oui		

Description

Le fichier ICS joint provoque un crash lorsqu'on tente de l'importer comme source d'exception sur un agenda

```
Date: Thu, 11 Oct 2018 13:16:59 -0000
From: root@chrono.rbx.tst.entrouvert.org
To: admin+chrono.test@entrouvert.com
Subject: [agendas-atreal.test.entrouvert.org] ERROR (EXTERNAL IP): Internal Server Error: /manage/agendas/desk/10/import-exceptions-from-ics/
```

```
Internal Server Error: /manage/agendas/desk/10/import-exceptions-from-ics/
Traceback (most recent call last):
  File "/usr/lib/python2.7/dist-packages/django/core/handlers/base.py", line 132, in get_response
    response = wrapped_callback(request, *callback_args, **callback_kwargs)
  File "/usr/lib/python2.7/dist-packages/django/contrib/auth/decorators.py", line 22, in _wrapped_view
    return view_func(request, *args, **kwargs)
  File "/usr/lib/python2.7/dist-packages/django/views/generic/base.py", line 71, in view
    return self.dispatch(request, *args, **kwargs)
  File "/usr/lib/python2.7/dist-packages/chrono/manager/views.py", line 449, in dispatch
    return super(ManagedAgendaSubobjectMixin, self).dispatch(request, *args, **kwargs)
  File "/usr/lib/python2.7/dist-packages/django/views/generic/base.py", line 89, in dispatch
    return handler(request, *args, **kwargs)
  File "/usr/lib/python2.7/dist-packages/django/views/generic/edit.py", line 272, in post
    return super(BaseUpdateView, self).post(request, *args, **kwargs)
  File "/usr/lib/python2.7/dist-packages/django/views/generic/edit.py", line 215, in post
    return self.form_valid(form)
  File "/usr/lib/python2.7/dist-packages/chrono/manager/views.py", line 752, in form_valid
    exceptions = form.instance.create_timeperiod_exceptions_from_ics(ics_file_content)
  File "/usr/lib/python2.7/dist-packages/chrono/agendas/models.py", line 546, in create_timeperiod_exceptions_from_ics
    if not is_aware(vevent.rruleset[0]):
  File "/usr/lib/python2.7/dist-packages/dateutil/rrule.py", line 148, in __getitem__
    raise IndexError
IndexError
```

```
Request repr():
<WSGIRequest
path:/manage/agendas/desk/10/import-exceptions-from-ics/,
...>
```

Demandes liées:

Lié à Chrono - Development #29209: matérialiser les sources d'exceptions Fermé 19 décembre 2018

Révisions associées

Révision b364e118 - 22 octobre 2018 12:08 - Benjamin Dauvergne

test rruleset emptyness (#27263)

ruleset does not define a *nonzero* magic method like a container would do.

Révision 07402527 - 22 octobre 2018 12:08 - Benjamin Dauvergne

improve filter for ponctual vevent exceptions (#27263)

To differentiate events between those coming from locally uploaded and remotely fetched .ics files, we must look up the locally uploaded ones with a filter of "recurrence_id=0".

Historique

#2 - 12 octobre 2018 10:08 - Benjamin Dauvergne

- Assigné à mis à Benjamin Dauvergne

Je prends.

#3 - 12 octobre 2018 10:12 - Benjamin Dauvergne

- Fichier 0001-test-rruleset-emptyness-using-len-fixes-27263.patch ajouté

- Statut changé de Nouveau à Solution proposée

- Patch proposed changé de Non à Oui

#4 - 12 octobre 2018 10:20 - Frédéric Péters

Je regarde le getrruleset sur ma copie de vobject et elle me semble pouvoir retourner None, ce qui échouerait ici.

Ajouter dans test_timeperiodexception_creation_from_ics_with_reurrences un passage sur un ics qui faisait planter ?

(aussi, je préfère éviter les (fixes: #...) dans les messages de commit (qui font faire des trucs automatiquement pas toujours raisonnables))

#5 - 12 octobre 2018 11:44 - Benjamin Dauvergne

Frédéric Péters a écrit :

Je regarde le getrruleset sur ma copie de vobject et elle me semble pouvoir retourner None, ce qui échouerait ici.

Oui tout à fait, en plus il n'y a pas de `len()` non plus apparemment, il faut faire `.count()` j'ai fait nawak je reprends.

Ajouter dans test_timeperiodexception_creation_from_ics_with_reurrences un passage sur un ics qui faisait planter ?

Yep.

(aussi, je préfère éviter les (fixes: #...) dans les messages de commit (qui font faire des trucs automatiquement pas toujours raisonnables))

Bon à la correction je me rends compte qu'il y a un problème sous-jacent plus embêtant, là pour l'instant mon fix qui tourne avec le fichier transmis par atreal:

```
diff --git a/chrono/agendas/models.py b/chrono/agendas/models.py
index fc09bb2..f414c26 100644
--- a/chrono/agendas/models.py
+++ b/chrono/agendas/models.py
@@ -529,6 +529,7 @@ class Desk(models.Model):
+
+         kwargs = {}
+         kwargs['desk'] = self
+         kwargs['recurrence_id'] = 0
+         if keep_synced_by_uid:
+             kwargs['external_id'] = vevent.contents['uid'][0].value
+         else:
@@ -539,7 +540,7 @@ class Desk(models.Model):
+         obj, created = TimePeriodException.objects.update_or_create(defaults=event, **kwargs)
+         if created:
+             total_created += 1
-         else:
+         elif vevent.rruleset.count():
+             # recurring event until recurring_days in the future
+             from_dt = start_dt
+             until_dt = update_datetime + datetime.timedelta(days=recurring_days)
```

Le premier changement c'est parce que beaucoup de leur événements dans leur fichier .ics ont le même summary (SUMMARY:X3M) et dans le cas de l'import d'un fichier .ics on utilise le summary comme label et le label comme clé, dans le cas d'un import distant on utilise le UID comme external_id et external_id comme clé; de plus on nettoie en sortie les UID devenus inconnus (et les recurrence_id supérieurs au dernier créé pour chaque external_id, enfin chaque VEVENT), donc là l'idée pour une récurrence qui génère le même start/end qu'un autre évènement et avec le

même label c'est de spécifier recurrence_id à 0 comme clé par défaut (recurrence_id a pour valeur par défaut 0 donc ça roule).

Le deuxième c'est pour le problème soulevé par la trace, un VEVENT avec une RRULE dont le UNTIL est avant DTSTART (donc récurrence vide).

Leur fichier est long dans mon test je ne valide rien de ce qui est chargé (ça crée 87 exceptions), est-ce que je réduis le fichier au VEVENT qui pose problème ? Ça ne couvrira plus le problème corrigé par mon premier "hunk".

#6 - 12 octobre 2018 11:45 - Benjamin Dauvergne

- Fichier 0002-improve-filter-for-ponctual-vevent-exceptions.patch ajouté

- Fichier 0001-test-rruleset-emptyness-27263.patch ajouté

La série à jour.

#7 - 12 octobre 2018 12:15 - Frédéric Péters

Le premier changement c'est parce que beaucoup de leur évènements dans leur fichier .ics ont le même summary (SUMMARY:X3M) et dans le cas de l'import d'un fichier .ics on utilise le summary comme label et le label comme clé

Il y a [#25962](#) sur le sujet.

#8 - 12 octobre 2018 17:02 - Benjamin Dauvergne

Frédéric Péters a écrit :

Le premier changement c'est parce que beaucoup de leur évènements dans leur fichier .ics ont le même summary (SUMMARY:X3M) et dans le cas de l'import d'un fichier .ics on utilise le summary comme label et le label comme clé

Il y a [#25962](#) sur le sujet.

Sans cette modif le fichier en question va continuer à cracher au chargement, l'autre ticket c'est un changement de comportement, ici on conserve le comportement actuel mais au moins on ne crashe pas.

#9 - 22 octobre 2018 10:29 - Frédéric Péters

- Statut changé de Solution proposée à Solution validée

Ok.

#10 - 22 octobre 2018 12:11 - Benjamin Dauvergne

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

```
commit 07402527c2002f9636243886666741602f508b8f (HEAD -> master, origin/master, origin/HEAD)
Author: Benjamin Dauvergne <bdauvergne@entrouvert.com>
Date: Fri Oct 12 11:44:54 2018 +0200
```

```
improve filter for ponctual vevent exceptions (#27263)
```

```
To differentiate events between those coming from locally uploaded and remotely fetched
.ics files, we must look up the locally uploaded ones with a filter of
"recurrence_id=0".
```

```
commit b364e11855437dac54f74610613b5aef8d1e4d4e
Author: Benjamin Dauvergne <bdauvergne@entrouvert.com>
Date: Fri Oct 12 11:06:42 2018 +0200
```

```
test rruleset emptyness (#27263)
```

```
rruleset does not define a __nonzero__ magic method like a container
would do.
```

#11 - 19 décembre 2018 12:37 - Benjamin Dauvergne

- Lié à Development #29209: matérialiser les sources d'exceptions ajouté

#12 - 23 décembre 2018 15:25 - Frédéric Péters

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

Fichiers

sbon@atreal.fr.ics	48,7 ko	11 octobre 2018	Thomas Noël
0001-test-rruleset-emptyness-using-len-fixes-27263.patch	948 octets	12 octobre 2018	Benjamin Dauvergne
0002-improve-filter-for-ponctual-vevent-exceptions.patch	769 octets	12 octobre 2018	Benjamin Dauvergne
0001-test-rruleset-emptyness-27263.patch	52 ko	12 octobre 2018	Benjamin Dauvergne