

Chrono - Bug #46145

Les créneaux d'un guichet ne sont pas vus dans via l'API (dans certaines conditions non éclaircies)

27 août 2020 16:43 - Emmanuel Cazenave

Statut:	Fermé	Début:	27 août 2020
Priorité:	Normal	Echéance:	
Assigné à:	Emmanuel Cazenave	% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	Non
Patch proposed:	Oui		

Description

C'est #46115 où les créneaux d'un guichet (le guichet 5) passent à la trappe sur le endpoint datetime.

Il y a un bout de code qui foire dans `get_all_slots` :

```
desks_by_min_max_datetime = {
    datetime_range: list(desks)
    for datetime_range, desks in itertools.groupby(
        time_period.desks,
        key=lambda desk: desk_min_max_datetime.get(desk, agenda_id_min_max_datetime[desk.agenda_id
    ]),
    )
}
```

Avec dans les conditions du ticket client le guichet 5 qui passe à la trappe dans `desks_by_min_max_datetime` :

```
(Pdb) time_period.desks
{<Desk: Guichet 5>, <Desk: Guichet 6>, <Desk: Guichet 1>, <Desk: Guichet 2>, <Desk: Guichet 3>, <D
esk: Guichet 4>}
```

```
(Pdb) desk_min_max_datetime
{<Desk: Guichet 1>: (datetime.datetime(2020, 8, 27, 0, 0, tzinfo=<UTC>), datetime.datetime(2020, 1
1, 1, 0, 0, tzinfo=<UTC>)), <Desk: Guichet 2>: (datetime.datetime(2020, 8, 27, 0, 0, tzinfo=<UTC>
), datetime.datetime(2020, 11, 1, 0, 0, tzinfo=<UTC>)), <Desk: Guichet 3>: (datetime.datetime(2020,
8, 27, 0, 0, tzinfo=<UTC>), datetime.datetime(2020, 10, 2, 0, 0, tzinfo=<UTC>)), <Desk: Guichet 4
>: (datetime.datetime(2020, 8, 27, 0, 0, tzinfo=<UTC>), datetime.datetime(2021, 2, 24, 0, 0, tzinf
o=<UTC>)), <Desk: Guichet 5>: (datetime.datetime(2020, 8, 27, 0, 0, tzinfo=<UTC>), datetime.dateti
me(2021, 2, 24, 0, 0, tzinfo=<UTC>)), <Desk:
Guichet 6>: (datetime.datetime(2020, 8, 27, 0, 0, tzinfo=<UTC>), datetime.datetime(2020, 9, 3, 0,
0, tzinfo=<UTC>))}
```

```
(Pdb) desks_by_min_max_datetime
{(datetime.datetime(2020, 8, 27, 0, 0, tzinfo=<UTC>), datetime.datetime(2021, 2, 24, 0, 0, tzinfo=
<UTC>)): [<Desk: Guichet 4>], (datetime.datetime(2020, 8, 27, 0, 0, tzinfo=<UTC>), datetime.dateti
me(2020, 9, 3,
0, 0, tzinfo=<UTC>)): [<Desk: Guichet 6>], (datetime.datetime(2020, 8, 27, 0, 0, tzinfo=<UTC>), da
tetime.datetime(2020, 11, 1, 0, 0, tzinfo=<UTC>)): [<Desk: Guichet 1>, <Desk: Guichet 2>], (dateti
me.datetime(2020, 8, 27, 0, 0, tzinfo=<UTC>), datetime.datetime(2020, 10, 2, 0, 0, tzinfo=<UTC>)):
 [<Desk: Guichet 3>]}
```

S'ensuit une itération sur `desks_by_min_max_datetime` d'où le guichet 5 est absent, etc.

Révisions associées

Révision 38df08da - 03 septembre 2020 10:22 - Emmanuel Cazenave

api: use a for loop instead of itertools.group_by (#46145)

Historique

#1 - 27 août 2020 16:43 - Emmanuel Cazenave

- Statut changé de Nouveau à En cours

- Assigné à mis à Emmanuel Cazenave

#2 - 27 août 2020 17:08 - Emmanuel Cazenave

- Fichier 0001-api-use-a-for-loop-instead-of-itertools.groupby-461.patch ajouté
- Statut changé de En cours à Solution proposée
- Patch proposed changé de Non à Oui

Je n'arrive pas à reproduire proprement dans un test unitaire, les conditions à réunir m'échappent.

Mais d'un script posé sur la prod avec le patch suivant appliqué, les slots du guichet 5 remontent bien via `get_all_slots`. Avec dans le détail à comparer à la description :

```
(Pdb) time_period.desks
{<Desk: Guichet 5>, <Desk: Guichet 6>, <Desk: Guichet 1>, <Desk: Guichet 2>, <Desk: Guichet 3>, <Desk: Guichet 4>}
```

```
(Pdb) desk_min_max_datetime
{<Desk: Guichet 1>: (datetime.datetime(2020, 8, 27, 0, 0, tzinfo=<UTC>), datetime.datetime(2020, 11, 1, 0, 0, tzinfo=<UTC>)), <Desk: Guichet 2>: (datetime.datetime(2020, 8, 27, 0, 0, tzinfo=<UTC>), datetime.datetime(2020, 11, 1, 0, 0, tzinfo=<UTC>)), <Desk: Guichet 3>: (datetime.datetime(2020, 8, 27, 0, 0, tzinfo=<UTC>), datetime.datetime(2020, 10, 2, 0, 0, tzinfo=<UTC>)), <Desk: Guichet 4>: (datetime.datetime(2020, 8, 27, 0, 0, tzinfo=<UTC>), datetime.datetime(2021, 2, 24, 0, 0, tzinfo=<UTC>)), <Desk: Guichet 5>: (datetime.datetime(2020, 8, 27, 0, 0, tzinfo=<UTC>), datetime.datetime(2021, 2, 24, 0, 0, tzinfo=<UTC>)), <Desk: Guichet 6>: (datetime.datetime(2020, 8, 27, 0, 0, tzinfo=<UTC>), datetime.datetime(2020, 9, 3, 0, 0, tzinfo=<UTC>))}
```

```
(Pdb) desks_by_min_max_datetime
defaultdict(<class 'list'>, {(datetime.datetime(2020, 8, 27, 0, 0, tzinfo=<UTC>), datetime.datetime(2021, 2, 24, 0, 0, tzinfo=<UTC>)): [<Desk: Guichet 5>, <Desk: Guichet 4>], (datetime.datetime(2020, 8, 27, 0, 0, tzinfo=<UTC>), datetime.datetime(2020, 9, 3, 0, 0, tzinfo=<UTC>)): [<Desk: Guichet 6>], (datetime.datetime(2020, 8, 27, 0, 0, tzinfo=<UTC>), datetime.datetime(2020, 11, 1, 0, 0, tzinfo=<UTC>)): [<Desk: Guichet 1>, <Desk: Guichet 2>], (datetime.datetime(2020, 8, 27, 0, 0, tzinfo=<UTC>), datetime.datetime(2020, 10, 2, 0, 0, tzinfo=<UTC>)): [<Desk: Guichet 3>]})
```

#4 - 27 août 2020 17:17 - Frédéric Péters

Il y a un deuxième appel à `itertools.groupby()` plus loin; évidemment sans les conditions de reproduction c'est compliqué d'imaginer si ça peut foirer pareil là.

#5 - 27 août 2020 17:37 - Emmanuel Cazenave

De <https://docs.python.org/3/library/itertools.html#itertools.groupby> : "Generally, the iterable needs to already be sorted on the same key function."

Dans mon exemple `time_period.desks` n'est pas trié par la 'key function', j'imagine que c'est ce qui pose problème. Le tri préalable me semble correct sur les deux `groupby` qui restent.

Je n'ai pas cherché à trier parce que je trouve ma version plus simple à comprendre.

#6 - 27 août 2020 17:41 - Benjamin Dauvergne

C'est l'absence de tri sur `time_period.desks` par `desk_min_max_datetime` qui casse le fonctionnement de `groupby`, les autres `groupby` sont triés; je pense donc que ta correction est suffisante.

#7 - 27 août 2020 17:43 - Benjamin Dauvergne

Emmanuel Cazenave a écrit :

Je n'ai pas cherché à trier parce que je trouve ma version plus simple à comprendre.

Oui.

#8 - 03 septembre 2020 10:20 - Christophe Siraut

- Statut changé de Solution proposée à Solution validée

#9 - 03 septembre 2020 10:23 - Emmanuel Cazenave

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

commit 38df08daf06b05e88e2211ca14a51f52c25b3f3d

Author: Emmanuel Cazenave <ecazenave@entrouvert.com>

Date: Thu Aug 27 16:46:13 2020 +0200

```
api: use a for loop instead of itertools.groupby (#46145)
```

#10 - 03 septembre 2020 17:16 - Frédéric Péters

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

Fichiers

0001-api-use-a-for-loop-instead-of-itertools.groupby-461.patch	1,37 ko	27 août 2020	Emmanuel Cazenave
--	---------	--------------	-------------------