

# Chrono - Bug #20732

## Trace chrono meaux

15 décembre 2017 10:20 - Josué Kouka

<b>Statut:</b>	Fermé	<b>Début:</b>	15 décembre 2017
<b>Priorité:</b>	Normal	<b>Echéance:</b>	
<b>Assigné à:</b>	Frédéric Péters	<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		

**Description**

```
Internal Server Error: /api/agenda/agenda1/meetings/3pers/datetimes/
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/views/decorators/csrf.py", line 58, in wrapped_vie
w
    return view_func(*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/rest_framework/views.py", line 466, in dispatch
    response = self.handle_exception(exc)
  File "/usr/lib/python2.7/dist-packages/rest_framework/views.py", line 463, in dispatch
    response = handler(request, *args, **kwargs)
  File "/usr/lib/python2.7/dist-packages/chrono/api/views.py", line 210, in get
    open_slots, all_time_slots = get_all_slots(agenda, meeting_type)
  File "/usr/lib/python2.7/dist-packages/chrono/api/views.py", line 83, in get_all_slots
    open_slots_by_desk[event.desk_id].remove_overlap(event.start_datetime, event.end_datetime)
  File "/usr/lib/python2.7/dist-packages/intervaltree/intervaltree.py", line 473, in remove_overla
p
    self.remove(iv)
  File "/usr/lib/python2.7/dist-packages/intervaltree/intervaltree.py", line 361, in remove
    self.top_node = self.top_node.remove(interval)
  File "/usr/lib/python2.7/dist-packages/intervaltree/node.py", line 211, in remove
    return self.remove_interval_helper(interval, done, should_raise_error=True)
  File "/usr/lib/python2.7/dist-packages/intervaltree/node.py", line 270, in remove_interval_helpe
r
    self[direction] = self[direction].remove_interval_helper(interval, done, should_raise_error)
  File "/usr/lib/python2.7/dist-packages/intervaltree/node.py", line 248, in remove_interval_helpe
r
    raise KeyError(interval)
KeyError: Interval(datetime.datetime(2017, 12, 22, 15, 10, tzinfo=<DstTzInfo 'Europe/Paris' CET+1:00:00 STD>), datetime.datetime(2017, 12, 22, 16, 10, tzinfo=<DstTzInfo 'Europe/Paris' CET+1:00:00 STD>), <chrono.agendas.models.TimeSlot object at 0x7f2b08f98310>)

Request repr():
<WSGIRequest
path:/api/agenda/agenda1/meetings/3pers/datetimes/,
GET:<QueryDict: {u'nonce': [u'8a2c754c410bf7c2ff14dbcc6001fd5d'], u'timestamp': [u'2017-12-15T08:39:05Z'], u'signature': [u'Gkawvod0kDeWVie5qXBKwNnmjS3CDCEz/u9myy2JHQ0='], u'algo': [u'sha256'], u'orig': [u'eservices.mymeaux.fr']}>,
POST:<QueryDict: {}>,
COOKIES:{},
META:{u'CSRF_COOKIE': u'kPRrvo2dFP7zNq3EQFVPH1gwAB7wvg94',
'HTTP_ACCEPT': '*/*',
'HTTP_ACCEPT_ENCODING': 'gzip, deflate',
'HTTP_CONNECTION': 'close',
'HTTP_HOST': 'agendas.mymeaux.fr',
'HTTP_USER_AGENT': 'python-requests/2.4.3 CPython/2.7.9 Linux/2.6.32-48-pve',
'HTTP_X_FORWARDED_FOR': '5.135.221.5',
'HTTP_X_FORWARDED_PROTO': 'https',
'HTTP_X_FORWARDED_PROTOCOL': 'ssl',
```

```
'HTTP_X_FORWARDED_SSL': 'on',
'HTTP_X_REAL_IP': '5.135.221.5',
'PATH_INFO': 'u'/api/agenda/agenda1/meetings/3pers/datetimes/',
'QUERY_STRING': 'orig=eservices.mymeaux.fr&algo=sha256&timestamp=2017-12-15T08%3A39%3A05Z&nonce=8a2c754c410bf7c2ff14dbcc6001fd5d&signature=Gkawvod0kDeWVie5qXBKwNnmjS3CDCEz/u9myy2JHQ0%3D',
'RAW_URI': '/api/agenda/agenda1/meetings/3pers/datetimes/?orig=eservices.mymeaux.fr&algo=sha256&timestamp=2017-12-15T08%3A39%3A05Z&nonce=8a2c754c410bf7c2ff14dbcc6001fd5d&signature=Gkawvod0kDeWVie5qXBKwNnmjS3CDCEz/u9myy2JHQ0%3D',
'REMOTE_ADDR': '5.135.221.5',
'REQUEST_METHOD': 'GET',
'SCRIPT_NAME': 'u'',
'SERVER_NAME': 'agendas.mymeaux.fr',
'SERVER_PORT': '443',
'SERVER_PROTOCOL': 'HTTP/1.0',
'SERVER_SOFTWARE': 'unicorn/19.6.0',
'unicorn.socket': <socket._socketobject object at 0x7f2b09083d70>,
'wsgi.errors': <unicorn.http.wsgi.WSGIErrorsWrapper object at 0x7f2b088b7510>,
'wsgi.file_wrapper': <class 'unicorn.http.wsgi.FileWrapper'>,
'wsgi.input': <unicorn.http.body.Body object at 0x7f2b088b7710>,
'wsgi.multiprocess': True,
'wsgi.multithread': False,
'wsgi.run_once': False,
'wsgi.url_scheme': 'https',
'wsgi.version': (1, 0)>
```

## Révisions associées

### Révision 38b79a36 - 30 décembre 2017 08:56 - Benjamin Dauvergne

general: add custom implementation of interval sets (#20732)

### Révision 6d4afedc - 30 décembre 2017 08:56 - Benjamin Dauvergne

api: replace intervaltree by chrono.interval (#20732)

## Historique

### #1 - 15 décembre 2017 11:18 - Josué Kouka

ça a l'air d'être lié à un bug dans IntervalTree <https://github.com/chaimleib/intervaltree/issues/41>. J'essaie de m'en assurer.

### #2 - 15 décembre 2017 12:10 - Josué Kouka

Josué Kouka a écrit :

ça a l'air d'être lié à un bug dans IntervalTree <https://github.com/chaimleib/intervaltree/issues/41>. J'essaie de m'en assurer.

Bon je confirme que c'est bien ce bug. Pour ne pas que ça bloque sur la prod j'ai fait à ce que aucune exception ne soit levée lorsqu'un Interval à supprimer n'est pas trouvé en posant le "hotfix" ci dessous dans /usr/lib/python2.7/dist-packages/intervaltree/node.py

```
236         if self.center_hit(interval):
237             #if trace: print('Hit at {}'.format(self.x_center))
238
239             should_raise_error = False # hotfix
240             if not should_raise_error and interval not in self.s_center:
241                 done.append(1)
```

Pour m'assurer que tout marche, j'ai fait des tests via le shell django du tenant de meaux et via le formulaire mise en ligne sur la prod de Meaux (j'ai évidemment annulé mes rendez-vous).

Il y'a ce commit <https://github.com/chaimleib/intervaltree/commit/2d1b463db7e9dc81a74dc50c24c32fab4cb21d7f> d'il y'a 4 jours, mais il semble ne pas marcher dans la cas de meaux. Je continue de regarder coté de la lib pour voir si je peux soumettre un patch aussi...

### #3 - 15 décembre 2017 14:25 - Thomas Noël

Quelle situation qui n'est pas dans les tests de chrono produit ce bogue ? Selon moi il faut commencer par ajouter un test dans chrono qui lève le problème... non ?

### #4 - 15 décembre 2017 15:35 - Josué Kouka

Thomas Noël a écrit :

Quelle situation qui n'est pas dans les tests de chrono produit ce bogue ? Selon moi il faut commencer par ajouter un test dans chrono qui lève le problème... non ?

Yep vraie. J'essaie de sortir le jeu de tests en me basant sur les données de meaux.

#### #5 - 15 décembre 2017 15:37 - Josué Kouka

(En passant, le patch passant c'est <https://github.com/chaimleib/intervaltree/pull/63/commits/7898b6767b6e00f9ef856f2fb3831f7c98a469f4>)

#### #6 - 16 décembre 2017 05:40 - Benjamin Dauvergne

- Fichier 0001-add-a-custom-implementation-of-interval-sets.patch ajouté

- Fichier 0002-api-replace-intervaltree-by-chrono.interval.patch ajouté

Pour se passer d'intervaltree.

#### #7 - 17 décembre 2017 09:32 - Frédéric Péters

Pour se passer d'intervaltree.

L'approche me va bien mais il reste quand même à ajouter un test unitaire correspondant à la situation à Meaux et quelques tests pour porter la couverture de chrono/interval.py à 100% (quelques tests ou quelques suppressions de lignes, par exemple le constructeur d'Intervals n'est jamais appelé avec un paramètre).

#### #8 - 17 décembre 2017 20:53 - Frédéric Péters

Pendant ce temps...

```
Internal Server Error: /api/agenda/agenda2/meetings/3pers/datetimes/  
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/views/decorators/csrf.py", line 58, in wrapped_view  
    return view_func(*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/rest_framework/views.py", line 466, in dispatch  
    response = self.handle_exception(exc)  
  File "/usr/lib/python2.7/dist-packages/rest_framework/views.py", line 463, in dispatch  
    response = handler(request, *args, **kwargs)  
  File "/usr/lib/python2.7/dist-packages/chrono/api/views.py", line 210, in get  
    open_slots, all_time_slots = get_all_slots(agenda, meeting_type)  
  File "/usr/lib/python2.7/dist-packages/chrono/api/views.py", line 83, in get_all_slots  
    open_slots_by_desk[event.desk_id].remove_overlap(event.start_datetime, event.end_datetime)  
  File "/usr/lib/python2.7/dist-packages/intervaltree/intervaltree.py", line 473, in remove_overlap  
    self.remove(iv)  
  File "/usr/lib/python2.7/dist-packages/intervaltree/intervaltree.py", line 360, in remove  
    raise ValueError  
ValueError
```

Vient d'avoir lieu sur /api/agenda/agenda2/meetings/3pers/datetimes/, Josué ?

#### #9 - 18 décembre 2017 18:01 - Frédéric Péters

- Assigné à changé de Josué Kouka à Frédéric Péters

Je reprends, je vais faire le test nécessaire, puis j'intégrerai le travail de Benjamin, en y ajoutant au passage la couverture souhaitée.

#### #10 - 18 décembre 2017 22:04 - Frédéric Péters

- Fichier 0002-api-replace-intervaltree-by-chrono.interval-20732.patch ajouté

- Fichier 0003-tests-add-check-for-a-change-in-opening-hours.patch ajouté

- Fichier 0001-general-add-custom-implementation-of-interval-sets-2.patch ajouté

- Patch proposed changé de Non à Oui

À creuser, c'est vraiment totalement un bug spécifique au fonctionnement interne d'IntervalTree, même du côté de leur suivi de bugs pour reproduire ils n'ont rien de simple, juste ce fichier monstrueux : [https://github.com/chaimleib/intervaltree/files/694995/issue41a\\_test.txt](https://github.com/chaimleib/intervaltree/files/694995/issue41a_test.txt) Autant éviter une chose pareille ici. (i.e. ce n'est pas juste "retirer un intervalle qui n'existe pas")

Cela étant, sur des idées d'origine "simple" du soucis, j'ajoute un test sur un changement dans les heures d'ouverture et la vérification qu'une réservation qui désormais couvre la fin et le début d'un créneau bloque bien les deux.

Benjamin, concernant ton code, à regarder pour la couverture des tests je me rends compte qu'une part importante n'est pas utilisée du tout par Chrono (ce code vient d'un moment où c'était utile, ou c'était pour in fine à finir par construire un test de comparaison de perf avec intervaltree, ou ?).

Bref j'ai réduit à la portion utile à Chrono; ce ticket ne se perdra de toute façon pas, s'il y a des éléments à ajouter plus tard, on les retrouvera.

J'attache déjà le résultat mais je dois encore poser mes yeux attentivement sur le 0002, notamment réfléchir à la ligne :

```
slots = [slot for slot in slots if not slot.full and slot.start_datetime >= event.start_datetime and slot.end_datetime <= event.end_datetime]
```

Comme on demande à réserver à event.start\_datetime, et que tous les slots retournés doivent avoir une durée correcte, je me demande ce que je rate en pensant que ça pourrait être :

```
slots = [slot for slot in slots if not slot.full and slot.start_datetime == event.start_datetime]
```

(les tests passant correctement avec cette seconde version).

### #11 - 19 décembre 2017 12:10 - Benjamin Dauvergne

Frédéric Péters a écrit :

À creuser, c'est vraiment totalement un bug spécifique au fonctionnement interne d'IntervalTree, même du côté de leur suivi de bugs pour reproduire ils n'ont rien de simple, juste ce fichier monstrueux : [https://github.com/chaimleib/intervaltree/files/694995/issue41a\\_test.txt](https://github.com/chaimleib/intervaltree/files/694995/issue41a_test.txt) Autant éviter une chose pareille ici. (i.e. ce n'est pas juste "retirer un intervalle qui n'existe pas")

Cela étant, sur des idées d'origine "simple" du soucis, j'ajoute un test sur un changement dans les heures d'ouverture et la vérification qu'une réservation qui désormais couvre la fin et le début d'un créneau bloque bien les deux.

Benjamin, concernant ton code, à regarder pour la couverture des tests je me rends compte qu'une part importante n'est pas utilisée du tout par Chrono (ce code vient d'un moment où c'était utile, ou c'était pour in fine à finir par construire un test de comparaison de perf avec intervaltree, ou ?).

Un peu tout ça, au départ je voulais juste m'amuser à voir si je pouvais refaire la même chose qu'intervaltree sachant que je savais que nos besoin étaient bien moindre et que donc un algorithme plus simple était possible (en gros c'est 10x plus rapide qu'intervaltree, bien sûr je ne couvre pas tout le spectre ce qu'intervaltree fournit, ou éventuellement pas avec les mêmes performances).

Bref j'ai réduit à la portion utile à Chrono; ce ticket ne se perdra de toute façon pas, s'il y a des éléments à ajouter plus tard, on les retrouvera.

Pas de souci ça me va, je crois qu'à part remove\_overlap pas grand chose ne sert, par contre au passage il me semble que pas mal d'optimisation sont encore possible (Je ne sais pas si le listing des dates disponibles est un truc qui au niveau performance pose souci donc je n'ai rien fait), notamment il me semble que get\_all\_slots() pourrait directement recevoir l'intervalle final au lieu que ce soit refiltré ensuite, on devrait pouvoir requêter plus facilement les événements à retirer.

J'attache déjà le résultat mais je dois encore poser mes yeux attentivement sur le 0002, notamment réfléchir à la ligne :

[...]

Comme on demande à réserver à event.start\_datetime, et que tous les slots retournés doivent avoir une durée correcte, je me demande ce que je rate en pensant que ça pourrait être :

[...]

(les tests passant correctement avec cette seconde version).

Ça me semble correct mais j'ai encore un peu du mal entre les durées, les slots, les event et les timeperiod, donc je te ferai confiance tu connais mieux le modèle de donnée que moi. Pour moi le code est encore assez moche et pourrait être simplifié. Y ayant repensé depuis je pense qu'on pourrait même simplifier jusqu'au point où on a juste besoin d'une méthode intersect() et pas de passer par des IntervalSets dans get\_all\_slots() mais donc idée pour plus tard, idem si on met bien noir sur blanc que slot et event, exclusions etc.. ont des durée fixes dans une petite fourchette (15 minutes, 30 minutes, 1h, 2h, et c'est tou) jet donc sont posés sur une grille (heures divisés en quart d'heure) et pas arbitrairement sur l'axe des horaires on doit encore pouvoir simplifier le code et augmenter les performances (genre renvoyer tous les slots libre sur 6 mois bien bien vite), mais il me manque les définitions de base pour connaître les contraintes qui nous aideraient à simplifier les algos.

Dans mon code j'ai joué avec pytest-benchmark pas de souci sur le fait que ça dégage c'était juste pour me prouver que j'avais pas complètement planté le code par rapport à avant, mais ça pourrait être utile de connaître les perfs sur des exemples tordus (obtenir tous les slots sur 1 an pour des horaires compliqués et pleins d'exceptions) juste pour avoir une idée.

## #12 - 19 décembre 2017 15:46 - Frédéric Péters

Pour se donner du mou sur la prod, j'ai "corrigé" en restaurant le code intervaltree d'origine mais en lui évitant son bug, via l'ajout de :

```
for k, v in open_slots_by_desk.items():
    open_slots_by_desk[k] = v.copy()
```

Avant de lancer les remove\_overlap.

Testé sur les différentes URL de mymeaux.fr où j'ai vu passer des traces, et d'autres encore.

## #13 - 20 décembre 2017 10:08 - Frédéric Péters

Testé sur les différentes URL de mymeaux.fr où j'ai vu passer des traces, et d'autres encore.

Mais ce matin à nouveau une trace; vraisemblablement juste eu de la chance hier :/

## #14 - 20 décembre 2017 14:35 - Frédéric Péters

Déplacé mes deux lignes :

```
for k, v in open_slots_by_desk.items():
    open_slots_by_desk[k] = v.copy()
```

après la première série de remove\_overlap et ça m'apparaît tenir le coup.

## #15 - 30 décembre 2017 08:58 - Frédéric Péters

- Statut changé de *En cours* à *Résolu* (à déployer)

J'ai poussé l'affaire, que ça puisse passer du temps sur la recette.

```
commit 6d4afedc475d979c531da869d392888e9b07ee76
Author: Benjamin Dauvergne <bdauvergne@entrouvert.com>
Date: Sat Dec 16 05:37:00 2017 +0100
```

```
api: replace intervaltree by chrono.interval (#20732)
```

```
commit 38b79a368c199961ed17bblce3b3bbb356d43f50
Author: Benjamin Dauvergne <bdauvergne@entrouvert.com>
Date: Sat Dec 16 04:17:54 2017 +0100
```

```
general: add custom implementation of interval sets (#20732)
```

## #16 - 23 décembre 2018 15:27 - Frédéric Péters

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

### Fichiers

0001-add-a-custom-implementation-of-interval-sets.patch	11,3 ko	16 décembre 2017	Benjamin Dauvergne
0002-api-replace-intervaltree-by-chrono.interval.patch	6,09 ko	16 décembre 2017	Benjamin Dauvergne
0002-api-replace-intervaltree-by-chrono.interval-20732.patch	6,1 ko	18 décembre 2017	Frédéric Péters
0003-tests-add-check-for-a-change-in-opening-hours.patch	1,93 ko	18 décembre 2017	Frédéric Péters
0001-general-add-custom-implementation-of-interval-sets-2.patch	6,83 ko	18 décembre 2017	Frédéric Péters