

Chrono - Development #37123

Agendas virtuels

22 octobre 2019 10:19 - Frédéric Péters

Statut:	Fermé	Début:	22 octobre 2019
Priorité:	Normal	Echéance:	31 mars 2020
Assigné à:	Emmanuel Cazenave	% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	Non
Patch proposed:	Oui		
Description			
<p>À côté d'agenda type rendez-vous et d'agenda type événement, avoir un nouveau type, "méta rendez-vous" (idéalement avec un nom plus clair), dont la définition sera une série d'agendas du même type (rendez-vous ou événement, pas de combinaison).</p> <p>Ça permet ainsi, pour ce qui est utilisé côté w.c.s., de ne pas avoir de changement, https://.../api/agenda/XXX/datetimes/, et sur un appel alors il y a d'un côté agrégation des créneaux/événements (pour /datetimes/), et de l'autre côté piochage du créneau/événement dans un des agendas (pour /fillslot/).</p> <p>On peut ainsi également gagner la vue backoffice, permettant à un agent de visualiser les rendez-vous/événements de plusieurs agendas à la fois.</p> <p>On pourra au niveau du "méta agenda" définir des éléments supplémentaires de configuration, selon les situations cela permettra différentes choses :</p> <ul style="list-style-type: none">• avoir pour une même série d'événements des réservations par les usagers jusqu'à l'avant-veille (les usagers utiliseraient l'agenda réel), et par les agents jusqu'au jour même (les agents utiliseraient le méta-agenda configuré sans limite de date de réservation).• pour un agenda de type rendez-vous y définir ses propres zones d'absence, ainsi en frontoffice on taperait sur un méta-agenda avec une absence définie le vendredi, alors qu'en backoffice on taperait sur un autre méta-agenda, basé sur les mêmes agendas, mais sans absence particulière définie.• pour un agenda de type rendez-vous également, définir la politique de remplissage des créneaux (concentrer vs diluer) (concentrer = remplir au maximum le même agenda/guichet, diluer = répartir les rendez-vous sur les différents agendas/guichets).			
Demandes liées:			
Lié à Chrono - Development #37120: Pouvoir différencier le délai de réservati...		Nouveau	22 octobre 2019
Lié à Publik - Development #37542: Création d'un nouveau type d'agenda : agen...		Fermé	07 novembre 2019

Révisions associées

Révision 565d471d - 20 mars 2020 13:03 - Emmanuel Cazenave

start virtual agendas (#37123)

Révision e2f04151 - 20 mars 2020 13:03 - Emmanuel Cazenave

manager: add views for virtual agendas (#37123)

Révision a4666a8e - 20 mars 2020 13:07 - Frédéric Péters

misc: use dictionary substitution in translatable string (#37123)

Révision 7c079db2 - 20 mars 2020 14:30 - Frédéric Péters

misc: fix looking up for extra meeting types (#37123)

Historique

#1 - 22 octobre 2019 10:19 - Frédéric Péters

- Lié à Development #37120: Pouvoir différencier le délai de réservation par un agent et par un usager ajouté

#2 - 22 octobre 2019 11:17 - Pierre Cros

Si on trouve le latin plus clair que le grec ça peu être un "Super Agenda", Super Publik aime ça. C'est en tout cas une super fonctionnalité.

#3 - 18 novembre 2019 10:19 - Brice Mallet

- Lié à Development #37542: Création d'un nouveau type d'agenda : agenda virtuel ajouté

#4 - 27 novembre 2019 11:03 - Emmanuel Cazenave

Pierre Cros a écrit :

Si on trouve le latin plus clair que le grec ça peu être un "Super Agenda", Super Publik aime ça. C'est en tout cas une super fonctionnalité.

Je propose "agenda virtuel", qui marque bien le fait qu'aucun évènement n'y est directement attaché, que c'est une vue sur un/des agendas normaux, etc.

#5 - 06 décembre 2019 14:22 - Pierre Cros

Mince j'avais raté la proposition de Manu et j'ai dit meta-agenda au club. Mais Agenda virtuel je trouve ça bien. Je peux faire un mail sur la liste du club pour expliquer (si on retient ce nom).

#6 - 16 décembre 2019 14:31 - Emmanuel Cazenave

- Sujet changé de méta-agendas à Agendas virtuels

#9 - 21 janvier 2020 10:40 - Pierre Cros

- Echéance mis à 31 mars 2020

- Assigné à mis à Emmanuel Cazenave

Et le nom "Agenda virtuel" est acté auprès du département du Nord. Il sera communiqué au club lors de la sortie.

#10 - 06 février 2020 11:13 - Emmanuel Cazenave

- Statut changé de Nouveau à En cours

#11 - 12 février 2020 15:26 - Emmanuel Cazenave

- Fichier 0001-start-virtual-agendas-37123.patch ajouté

- Statut changé de En cours à Solution proposée

- Patch proposed changé de Non à Oui

Un patch pour "early review", manque le boulot sur l'UI (à suivre dans ce ticket).

Et aussi proposition minimaliste : pas de politique de remplissage intelligente, pas de gestion d'exception temporelles posées sur l'agenda virtuel, restriction de la "virtualisation" aux agendas de type rdv (tout ça me paraît pouvoir être fait dans d'autres tickets, l'ordre mentionné me paraissant être l'ordre de priorité).

Un agenda virtuel expose des types de rendez vous : uniquement ceux qui sont partagés par tous ses agendas réels (même slug, label, durée) : Agenda.iter_meetingtypes. Ces types de rendez vous sont des objets transient et donc nécessité de pouvoir les retrouver via leur slug : Agenda.get_meetingtype, TimeSlot.__init__.

Une fois cela posé ça devient assez simple, pour exposer les créneaux, il y a juste à rajouter des itérations sur les agendas réels dans get_all_slots. Et quasiment rien à faire pour poser une réservation (puisque pour l'instant on remplit bêtement).

Sur les types de rendez-vous, "uniquement ceux qui sont partagés par tous ses agendas réels", ça a un coté fragile, genre un label changé sur un type de rendez vous d'un agenda réel et plouf l'agenda virtuel n'expose plus aucun type de rendez vous : je pensais mettre des gardes fous dans l'UI, genre attention si tu changes ce label l'agenda virtuel toto devient inutilisable.

#12 - 12 février 2020 15:26 - Emmanuel Cazenave

- Statut changé de Solution proposée à En cours

#13 - 20 février 2020 16:13 - Emmanuel Cazenave

- Fichier 0001-start-virtual-agendas-37123.patch ajouté

- Fichier 0002-manager-add-views-for-virtual-agendas-37123.patch ajouté

- Statut changé de En cours à Solution proposée

La suite donc avec le boulot sur le manager (démon : <https://perso.entrouvert.org/~ecazenave/virtual-agendas-sound.ogg>)

Dans la série petits pas, sur la vue de base d'un agenda je zappe la partie affichage des réservations pour n'afficher que des liens vers les agendas réels.

#14 - 24 février 2020 16:48 - Emmanuel Cazenave

- Fichier 0001-start-virtual-agendas-37123.patch ajouté

- Fichier 0002-manager-add-views-for-virtual-agendas-37123.patch ajouté

Des brouilles corrigées.

#15 - 24 février 2020 17:38 - Emmanuel Cazenave

- Fichier 0001-start-virtual-agendas-37123.patch ajouté

- Fichier 0002-manager-add-views-for-virtual-agendas-37123.patch ajouté

Idem.

#16 - 02 mars 2020 18:45 - Frédéric Péters

Commentaire général d'une première lecture de l'ensemble des commits,

```
+def compute_effective_timeperiods(effective_timeperiods, excluded_timeperiods):
```

J'y aurais plutôt vu une boucle qu'une récursion, mais ok.

```
+ def accept_meetings(self):
+     if self.kind == 'virtual':
+         return not self.real_agendas.filter(kind='events').exists()
```

Pourquoi pas un `self.real_agendas.filter(kind='meetings').exists()`, sans la négation, qui me semblerait ne pas faire prendre de risque aux futures évolutions qui ajouteraient d'autres types d'agenda ?

```
+     On a virtual agenda we expose transient meeting types based on on the
+     the real ones shared by every real agendas.
```

Ça me semble la décision importante ici; comment tu arrives à ça plutôt que prendre la totalité des types de rendez-vous ?

```
+     # select the agenda a desk on the agenda with min fill_rate on the given date
```

(retirer le premier "the agenda") mais aussi déjà créer un ticket pour prévoir une préférence de remplissage (répartir en mettant là où il y a le moins de rendez-vous, comme ici, ou à l'inverse remplir au maximum un guichet avant d'«ouvrir» le suivant).

#17 - 02 mars 2020 21:18 - Emmanuel Cazenave

Frédéric Péters a écrit :

```
+ On a virtual agenda we expose transient meeting types based on on the
+     the real ones shared by every real agendas.
```

Ça me semble la décision importante ici; comment tu arrives à ça plutôt que prendre la totalité des types de rendez-vous ?

Oui c'est clairement le plus important et ce dont on doit parler.

Sur les appels de listing de type de rendez-vous `{{agendas_url}}api/agenda/IDENTIFIANT-AGENDA/meetings/`, il me semble obligatoire d'agréger pour que ça puisse conserver un sens lorsque c'est présenté dans un formulaire.

Sur les appels de disponibilité des créneaux `{{agendas_url}}api/agenda/RENDEZ-VOUS-PASSEPORT/meetings/TYPE-DE-RDV/datetimes/`, le type de TYPE-DE-RDV il représente aussi plusieurs MeetingType réels.

Et agréger uniquement sur la base du slug par exemple, ça m'a semblé trop léger, que ça ouvrait trop la porte au mélange patate/carotte involontaire, donc j'ai pris l'option même slug, même durée, même label. Sur le cas d'usage du Nord et des rendez-vous de travailleurs sociaux, c'est raccord.

Coté implémentation sur les appels de disponibilité des créneaux

`{{agendas_url}}api/agenda/RENDEZ-VOUS-PASSEPORT/meetings/TYPE-DE-RDV/datetimes/`, ça me semblait simplifier de savoir qu'on a calculé des créneaux qui ont la même durée sur tous les agendas réels. A y regarder une deuxième fois c'est peut-être pas être pas vraiment fondé.

Aussi ça semblait intéressant de pouvoir compter sur cette hypothèse dans l'UI : quand tu inclues un agenda dans un agendas virtuels, je peux vérifier qu'il a le/les meeting types qui collent avec ce qui est déjà exposé par l'agenda virtuel, pour prévenir des situations du types j'ajoute un nouvel agenda dans un agenda virtuel et deux mois plus tard je me rends compte qu'aucun créneau n'est réservé, ben oui c'est parce qu'il a pas le même slug que les autres sur son/ses meeting type.

#18 - 16 mars 2020 17:47 - Emmanuel Cazenave

- Fichier 0001-start-virtual-agendas-37123.patch ajouté

- Fichier 0002-manager-add-views-for-virtual-agendas-37123.patch ajouté

Frédéric Péters a écrit :

Pourquoi pas un `self.real_agendas.filter(kind='meetings').exists()`, sans la négation, qui me semblerait ne pas faire prendre de risque aux futures évolutions qui ajouteraient d'autres types d'agenda ?

Encore plus radical :

```
return not self.real_agendas.filter(~Q(kind='meetings')).exists()
```

Pour le problème des types de rendez-vous, suite aux échanges sur liste etc, c'est maintenant en mode très restrictif, on ne peut ajouter un agenda dans un agenda virtuel que lorsque ceux ci partagent exactement les mêmes types de rdv (avec aussi des gardes fous sur ajout/édition/suppression d'un type de rendez-vous lorsque l'agenda est utilisé dans un agenda virtuel).

Et dans cette définition de "même type de rendez vous", j'ai finalement gardé la label. S'en passer pose un problème sur le endpoint d'API qui liste les types de rendez-vous (quel label exposer alors ?) et bien qu'il paraisse raisonnable à première vue d'être plus souple sur le label, on a pas de cas d'usage correspondant à ma connaissance.

#19 - 18 mars 2020 10:15 - Frédéric Péters

Je crée un agenda virtuel, je fais "include agenda", ça me dit "This agenda does not have the same meetingtypes provided by the virtual agenda."; il doit manquer un truc pour autoriser le premier agenda sans le comparer à quelque chose n'existant pas.

#20 - 18 mars 2020 15:12 - Emmanuel Cazenave

Oups désolé, j'avais pas refait un tour dans l'UI après les derniers changements, trop confiant dans mes tests qui ne couvraient pas cette situation.

C'est couvert maintenant et le fix en lui même consiste à donner la possibilité à un agenda virtuel d'exposer les types de rendez-vous qu'il aurait si un agenda X n'était pas inclu (la validation est faite dans le clean de VirtualMember, alors que le nouvel agenda est déjà ajouté; ça me sert aussi sur [#40057](#)) :

```
- def iter_meetingtypes(self):
+ def iter_meetingtypes(self, excluded_agenda=None):
    """ Expose agenda's meetingtypes.
    straightforward on a real agenda
    On a virtual agenda we expose transient meeting types based on on the
    the real ones shared by every real agendas.
    """
    if self.kind == 'virtual':
+         base_qs = MeetingType.objects.filter(agenda__virtual_agendas__in=[self])
+         real_agendas = self.real_agendas
+         if excluded_agenda:
+             base_qs = base_qs.exclude(agenda=excluded_agenda)
+             real_agendas = real_agendas.exclude(pk=excluded_agenda.pk)
        queryset = (
-         MeetingType.objects.filter(agenda__virtual_agendas__in=[self])
-         .values('slug', 'duration', 'label')
+         base_qs.values('slug', 'duration', 'label')
+         .annotate(total=Count('*'))
-         .filter(total=self.real_agendas.count())
+         .filter(total=real_agendas.count())
        )
    .
    .
-         virtual_meetingtypes = self.virtual_agenda.iter_meetingtypes()
+         virtual_meetingtypes = self.virtual_agenda.iter_meetingtypes(excluded_agenda=self.real_agenda)
-
-         if len(virtual_meetingtypes) != MeetingType.objects.filter(agenda=self.real_agenda).count():
+         num_virt_meetingtypes = len(virtual_meetingtypes)
+         if num_virt_meetingtypes and num_virt_meetingtypes != MeetingType.objects.filter(agenda=self.real_agenda).count():
```

Toutes les branches sont à jour.

#21 - 18 mars 2020 16:16 - Frédéric Péters

Dans le paramétrage d'un agenda virtuel, la liste des agendas inclus sont repris mais affichent [identifiant: whatever] plutôt que le nom de l'agenda. (à mon avis le nom est suffisant, pas besoin d'inclure le slug ici).

--

This agenda does not have the same meetingtypes provided by the virtual agenda.

"meeting types" espace.

Je pense que ce message va amener une certaine frustration quand il va falloir aller à la chasse à la différence pour comprendre pourquoi il s'affiche alors que tout a l'air identique; à réfléchir davantage mais peut-être avoir :

```
<!> This agenda does not have the same meeting types provided by the virtual agenda.
```

```
* Meeting type "foo" (30 minutes), no meeting with that label.
* Meeting type "bar" (60 minutes), meeting "bar" existing but with a different duration.
* Extra meeting type, "baz".
```

--

Deux mini-commentaires pour respirer,

```
+ # TODO VIRTUAL
```

Sa suppression a été oubliée dans le commit qui ajoute la prise en charge de l'export/import.

```
+ self.maximal_booking_delay = 56
```

Pour suivre Thomas et Benjamin dans [#40735](#), peut-être écrire 8 * 7 plutôt. (yep ça remplace un code où était écrit 56 avant)

--

Je réfléchissais aussi à l'affichage d'un agenda virtuel, en me disant qu'il y aurait à l'étendre avec quelques phrases de texte, mais que ça pourrait venir plus tard. Mais du coup j'avais les yeux sur le template :

```
{% for real_agenda in agenda.real_agendas.all %}
<li><a href="{% url 'chrono-manager-agenda-view' pk=real_agenda.pk %}">
  {{real_agenda.label}}
  <span class="identifiant">[ {% trans "identifiant:" %} {{real_agenda.slug}} ]</span>
</a>
{% endfor %}
```

et je vois qu'il ne prend pas en compte les paramètres d'accès, qu'on peut du coup avoir un lien vers une erreur si jamais un des agendas n'est pas accessible en lecture.

#22 - 18 mars 2020 19:15 - Emmanuel Cazenave

J'ai tenu compte de tout.

Sur les messages d'erreur je n'arrive à quelque chose d'aussi précis, mais avec tout de même en messages additionnels des :

```
Meeting type "Foo" (30 minutes) (identifiant: foo) does no exist.
Extra meeting type, "Bar".'
```

#23 - 20 mars 2020 09:03 - Frédéric Péters

- Statut changé de Solution proposée à Solution validée

Go ainsi. (cette validation vaut pour toute la branche).

#24 - 20 mars 2020 13:27 - Frédéric Péters

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

#25 - 20 mars 2020 13:28 - Frédéric Péters

J'ai tout poussé, j'ai du faire une modification de dernière minute pour les trads parce que

```
./agendas/models.py:307 : AVERTISSEMENT : La chaîne de format 'msgid' avec des arguments non nommés ne peut pas être correctement localisée :
```

```
Le traducteur ne peut pas réordonner les arguments.
Envisagez d'utiliser une chaîne de format contenant des arguments nommés,
et un mapping plutôt qu'un tuple pour les arguments.
```

#26 - 20 mars 2020 14:36 - Frédéric Péters

Autre commit fait pour faire ça, signalé par pylint :

```
-         slug=virt_meetingtype.slug, label=virt_meetingtype.label, duration=meetingtype.duration
+         slug=virt_meetingtype.slug, label=virt_meetingtype.label, duration=virt_meetingtype.durat
ion
```

Le coverage passait bien par là mais ça récupérait l'objet meetingtype d'une boucle précédente.

#27 - 20 mars 2020 15:15 - Frédéric Péters

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

Fichiers

0001-start-virtual-agendas-37123.patch	35 ko	12 février 2020	Emmanuel Cazenave
0001-start-virtual-agendas-37123.patch	37,6 ko	20 février 2020	Emmanuel Cazenave
0002-manager-add-views-for-virtual-agendas-37123.patch	22,4 ko	20 février 2020	Emmanuel Cazenave
0002-manager-add-views-for-virtual-agendas-37123.patch	23,7 ko	24 février 2020	Emmanuel Cazenave
0001-start-virtual-agendas-37123.patch	37,7 ko	24 février 2020	Emmanuel Cazenave
0001-start-virtual-agendas-37123.patch	37,7 ko	24 février 2020	Emmanuel Cazenave
0002-manager-add-views-for-virtual-agendas-37123.patch	23,8 ko	24 février 2020	Emmanuel Cazenave
0001-start-virtual-agendas-37123.patch	37,7 ko	16 mars 2020	Emmanuel Cazenave
0002-manager-add-views-for-virtual-agendas-37123.patch	29,2 ko	16 mars 2020	Emmanuel Cazenave