

## Chrono - Autre #42409

### Discussion: base\_meeting\_duration et agenda virtuel

03 mai 2020 11:20 - Benjamin Dauvergne

<b>Statut:</b>	Fermé	<b>Début:</b>	03 mai 2020
<b>Priorité:</b>	Normal	<b>Echéance:</b>	
<b>Assigné à:</b>		<b>% réalisé:</b>	0%
<b>Catégorie:</b>		<b>Temps estimé:</b>	0:00 heure
<b>Version cible:</b>		<b>Planning:</b>	Non
<b>Patch proposed:</b>	Non		

**Description**

base\_meeting\_duration est défini comme le plus grand diviseur commun des durées de rendez-vous pour un agenda.

Agenda.get\_base\_meeting\_duration() a été redéfini à l'occasion de l'introduction des agendas virtuels pour le calculer à partir des durées des rendez-vous communs de tous les agendas attachés à un agenda virtuel; mais cette version n'est jamais utilisée dans get\_all\_slots(), en effet get\_base\_meeting\_duration() n'est appelé<sup>1</sup> que sur les agendas réels.

Concrètement<sup>2</sup> cela veut dire que si on crée un agenda virtuel VA avec deux agendas réels RA1 et RA2 dont les durées de base des rendez-vous sont respectivement RA1.base\_meeting\_duration=30 minutes et RA2.base\_meeting\_duration=15 minutes (parce qu'ils existe un rendez-vous durant 15 minutes sur RA2) on aura pour V des rendez vous possibles toutes les 30 minutes sur les guichets de RA1/RA2 et toutes les 15 minutes uniquement sur les guichets de R2 (à supposer que leurs grilles d'ouverture, TimePeriod, soient identiques).

Je me demandais si c'était vraiment le résultat voulu, ou si on préférerait calculer base\_meeting\_duration uniquement sur VA (ça donnerait ici 30 minutes) ou comme le GCD des durées de base pour les agendas réels et là ça donnerait 5 minutes pour tous les guichets de RA1 et RA2 (mais donc des RdV seraient possibles via VA sur les guichets de RA1 qui ne seraient pas possible directement via R1, parce qu'on y applique une grille "plus fine").

<sup>1</sup><http://git.entrouvert.org/chrono.git/tree/chrono/api/views.py?id=9d0b76c82a3a0ac8b77613dba160070c1f4426ad#n92>

<sup>2</sup>

```
VA : Agenda = label="Virtual Agenda",kind="virtual",
get_base_meeting_duration = 30 (valeur jamais utilisée)
real_agendas =
  RA1 : Agenda = label="Real Agenda 1"
    get_base_meeting_duration() = 30
    mt1 = MeetingType : label="MT", duration=30
    desk1 : Desk = label="RA1 Desk"
      tp1 : TimePeriod = Lundi 10h -> 12h
  RA2 : Agenda : "Real Agenda 2"
    get_base_meeting_duration() = 15
    mt21 : MeetingType = label="MT", duration=30
    mt22 : MeetingType = label="Other", duration=15
    desk2 : Desk = label="RA2 Desk"
      tp2 : TimePeriod = Lundi 10h -> 12h

get_all_slots(agenda=VA, meetingtype="MT") :
  Lundi, 10h00 -> 10h30, desk=desk1
  Lundi, 10h00 -> 10h30, desk=desk2
  Lundi, 10h15 -> 10h45, desk=desk2 <- impossible pour desk1, à cause de base_meeting_duration
  Lundi, 10h30 -> 11h00, desk=desk1
  Lundi, 10h30 -> 11h00, desk=desk2
```

## Historique

#1 - 03 mai 2020 11:29 - Frédéric Péters

Concrètement cela veut dire que si on crée un agenda virtuel VA avec deux agendas réels RA1 et RA2 dont les durées de base des rendez-vous sont respectivement

Il me semblait que les agendas virtuels étaient uniquement possibles quand les rendez-vous définis des deux côtés étaient identiques et il y a quelques contrôles en place pour que ça soit le cas, tu arrives dans quelle situation à contourner ça ?

## #2 - 03 mai 2020 11:30 - Frédéric Péters

cf [#37123#note-18](#)

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).

## #3 - 03 mai 2020 13:29 - Benjamin Dauvergne

- Statut changé de Nouveau à Fermé

Hmm ok, comme pour l'autre ticket que j'avais ouvert sur le filtrage des RdV bookés ce sont des chose qui ne sont pas apparentes au niveau de la définition des modèles voir carrément contre-intuitive quand on lit l'algo de `get_all_slots()`, uniquement vérifiées dans les vues de gestion je suppose. Je vais voir pour documenter ça dans `get_all_slots()`, mais donc on peut encore simplifier le code `get_all_slots()` car on a l'assurance que `virtual_agenda.get_base_meeting_duration() == real.get_base_meeting_duration()` dans tous les cas.