

Combo - Bug #41276

Une cellule "liste de liens" est vide quand elle est dans une cellule "Identique à la page parent" (ParentContentCell)

02 avril 2020 16:48 - Thomas Noël

Statut:	Fermé	Début:	02 avril 2020
Priorité:	Normal	Echéance:	
Assigné à:	Frédéric Péters	% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	Non
Patch proposed:	Oui		

Description

Pour reproduire :

- sur une page, dans un placeholder, construire une cellule "Liste de liens" et y mettre 2 ou 3 liens
- sur une sous-page, poser une cellule "Identique à la page parent" dans le placeholder

dans la sous-page, la cellule "liste de liens" s'affiche mais avec une liste vide.

Révisions associées

Révision d66e3b55 - 14 avril 2020 17:47 - Frédéric Péters

misc: get cells from private placeholders in parent acquisition (#41276)

Historique

#1 - 02 avril 2020 16:52 - Thomas Noël

Je pense que c'est parce qu'on cherche les liens dans `for cell in context.get('page_cells', [])`: alors qu'on n'est plus sur la page de la cellule (celle où elle est instanciée).

#3 - 02 avril 2020 17:12 - Thomas Noël

J'ai l'impression que c'est juste ça, ou bien je rate une subtilité ?

```
diff --git a/combo/data/models.py b/combo/data/models.py
index a295707f..6f9581de 100644
--- a/combo/data/models.py
+++ b/combo/data/models.py
@@ -1241,11 +1241,7 @@ class LinkListCell(CellBase):
     def get_cell_extra_context(self, context):
         extra_context = super(LinkListCell, self).get_cell_extra_context(context)
         links = []
-        for cell in context.get('page_cells', []):
-            if not hasattr(cell, 'add_as_link_label'):
-                continue
-            if not cell.placeholder == self.link_placeholder:
-                continue
+        for cell in self.get_items():
+            links.append(cell.get_cell_extra_context(context))
         extra_context['links'] = links
         extra_context['more_links'] = []
```

#4 - 02 avril 2020 17:49 - Frédéric Péters

J'ai l'impression que c'est juste ça, ou bien je rate une subtilité ?

Minimiser le nombre de requêtes.

#5 - 02 avril 2020 17:54 - Thomas Noël

Frédéric Péters a écrit :

J'ai l'impression que c'est juste ça, ou bien je rate une subtilité ?

Minimiser le nombre de requêtes.

Yep.

Tenté sur <http://git.entrouvert.org/combo.git/?h=wip%2F41276-linklist-in-parentcell>

Et oui ça requête plus et donc le test « `assert len(ctx.captured_queries) == 31` » n'est pas content.

Ceci étant c'est a priori le seul pépin.

Si quelqu'un a une autre idée...

#6 - 02 avril 2020 18:01 - Frédéric Péters

Et là le test c'est sur la page du manager, donc moins important que le front côté nombre de requêtes. (mais modif sans effet vu que pas de problème côté manager).

Ce qu'il faut (d'un rapide regard) c'est adapter le `get_parent_cells()`, modifier

```
CellBase.get_cells(placeholder=self.placeholder, page__in=pages)
```

pour qu'il récupère aussi les cellules faisant partie de "liste de liens". (et une approximation genre `placeholder__startswith='_'` doit être ok, je pense).

#7 - 03 avril 2020 14:35 - Frédéric Péters

- Assigné à mis à Frédéric Péters

#8 - 03 avril 2020 15:11 - Frédéric Péters

- Fichier `0001-misc-get-cells-from-private-placeholders-in-parent-a.patch` ajouté

- Statut changé de Nouveau à Solution proposée

- Patch proposed changé de Non à Oui

Voilà, avec un moche mais efficace,

```
+         parent_cells = [seen.setdefault((x.__class__, x.id), True) and x
+         for x in parent_cells if (x.__class__, x.id) not in seen]
```

#9 - 03 avril 2020 16:04 - Thomas Noël

A priori le `seen` ajouté dans `combo/data/models.py` ne sert à rien (sans doute un reste d'idée ?)

Frédéric Péters a écrit :

Voilà, avec un moche mais efficace,

Si j'ai compris il s'agit de ne pas avoir de doublons. On (Lauréline et moi) a deux questions :

- c'est pour garder l'ordre que tu n'utilise pas un set ? Ou bien pour une question d'efficacité ?
- et aussi, pourquoi enlever les doublons ?

#10 - 03 avril 2020 16:06 - Lauréline Guérin

(et aussi le remplissage du `seen` dans la méthode `get_parents_cells` qui semble ne pas servir à grand chose :)

#11 - 03 avril 2020 16:37 - Frédéric Péters

A priori le `seen` ajouté dans `combo/data/models.py` ne sert à rien (sans doute un reste d'idée ?)

Yes, c'était dans l'idée de faire ça à cet endroit, mais ça ne marche pas.

Si j'ai compris il s'agit de ne pas avoir de doublons. On (Lauréline et moi) a deux questions :

c'est pour garder l'ordre que tu n'utilise pas un set ? Ou bien pour une question d'efficacité ?

Pour le `seen` ? C'est par habitude et certitude concernant les performances.

et aussi, pourquoi enlever les doublons ?

Quand on a une ParentContentCell, extend_with_parent_cells() est appelé pour étendre la liste des cellules figurant sur la page; l'appel faisait placeholder=self.placeholder et donc à étendre la zone "pied de page" on trouvait uniquement les cellules du pied de page, très bien.

À chercher également les cellules avec un placeholder commençant par un _, on attrape les "sous-cellules" (lien d'une cellule liste de liens, en l'occurrence).

Les doublons arrivent s'il y a une seconde ParentContentCell; elle va aussi faire le même job, récupérer les mêmes "sous-cellules", et celles-ci vont se trouver deux fois dans la liste des cellules figurant dans la page.

Ensuite, la cellule LinkListCell va faire

```
links = []
for cell in context.get('page_cells', []):
```

et si dans page_cells on avait laissé les cellules en double, elles seraient ici reprises deux fois.

#12 - 03 avril 2020 21:57 - Thomas Noël

- Statut changé de Solution proposée à Solution validée

Frédéric Péters a écrit :

Les doublons arrivent s'il y a une seconde ParentContentCell

J'ai tout pigé ; go.

#13 - 04 avril 2020 09:07 - Frédéric Péters

- Fichier 0001-misc-get-cells-from-private-placeholders-in-parent-a.patch ajouté

- Statut changé de Solution validée à Solution proposée

Je modifie un peu, pour filtrer sur des préfixes spécifiques, mon idée étant que sur une page comme "mon tableau de bord" le filtre _startswith= attraperait toutes les cellules des tableaux de bord de tous les usagers.

```
diff --git a/combo/data/models.py b/combo/data/models.py
index 35d5bb23..d5cd4219 100644
--- a/combo/data/models.py
+++ b/combo/data/models.py
@@ -569,6 +569,7 @@ class CellBase(six.with_metaclass(CellMeta, models.Model)):
     default_form_class = None
     manager_form_factory_kwargs = {}
     manager_form_template = 'combo/cell_form.html'
+    children_placeholder_prefix = None

     visible = True
     user_dependant = False
@@ -1213,6 +1214,7 @@ class LinkListCell(CellBase):

     template_name = 'combo/link-list-cell.html'
     manager_form_template = 'combo/manager/link-list-cell-form.html'
+    children_placeholder_prefix = '_linkslist:'

     invalid_reason_codes = {
         'data_link_invalid': _('Invalid link'),
@@ -1223,7 +1225,7 @@ class LinkListCell(CellBase):

     @property
     def link_placeholder(self):
-        return '_linkslist:{}'.format(self.pk)
+        return self.children_placeholder_prefix + str(self.pk)

     def get_items(self, prefetch_validity_info=False):
         return CellBase.get_cells(
@@ -1386,7 +1388,10 @@ class ParentContentCell(CellBase):
         cells_by_page[page.id] = []
         # get cells from placeholder + cells in private placeholders that may
         # be used by actual cells.
-        for cell in CellBase.get_cells(page__in=pages, extra_filter=Q(placeholder=self.placeholder) | Q(place
holder__startswith='_')):
+        placeholder_filter = Q(placeholder=self.placeholder)
```

```
+     for klass in CellBase.get_cell_classes(lambda x: bool(x.children_placeholder_prefix)):
+         placeholder_filter |= Q(placeholder__startswith=klass.children_placeholder_prefix)
+     for cell in CellBase.get_cells(page__in=pages, extra_filter=placeholder_filter):
+         cells_by_page[cell.page_id].append(cell)

cells = cells_by_page[pages[-1].id]
```

#14 - 14 avril 2020 17:44 - Thomas Noël

- *Statut changé de Solution proposée à Solution validée*

Je préférerais que `children_placeholder_prefix` ne contienne ni le "_" ni le ":"

```
children_placeholder_prefix = 'linklist'

...
@property
def link_placeholder(self):
    if children_placeholder_prefix:
        return '_{}:{}'.format(self.children_placeholder_prefix, self.pk)
```

et donc plus bas :

```
for klass in CellBase.get_cell_classes(lambda x: bool(x.children_placeholder_prefix)):
    placeholder_filter |= Q(placeholder__startswith='_' + klass.children_placeholder_prefix + ':')
```

et en écrivant ça je me dis que c'est pas joli. Laissons le `children_placeholder_prefix = '_linklist:'` mais pensons à respecter ce format pour les prochaines fois.

Sur ce `CellBase.get_cell_classes(lambda x: bool(x.children_placeholder_prefix))` je trouverai plus "lisible" d'écrire `CellBase.get_cell_classes(class_filter=lambda x: ...)` même si on ne le fait nulle part ailleurs.

Bref, allez, c'est un ack, fait.

#15 - 14 avril 2020 17:47 - Frédéric Péters

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

Poussé tel quel,

```
commit d66e3b55622bf48222561390668a08b5ebeab1b9
Author: Frédéric Péters <fpeters@entrouvert.com>
Date:   Fri Apr 3 14:43:58 2020 +0200
```

```
misc: get cells from private placeholders in parent acquisition (#41276)
```

#16 - 15 avril 2020 15:16 - Frédéric Péters

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

Fichiers

0001-misc-get-cells-from-private-placeholders-in-parent-a.patch	5,22 ko	03 avril 2020	Frédéric Péters
0001-misc-get-cells-from-private-placeholders-in-parent-a.patch	6,24 ko	04 avril 2020	Frédéric Péters