

Combo - Development #21052

performance : divers

09 janvier 2018 10:31 - Frédéric Péters

| | | | |
|------------------------|-----------------|----------------------|-----------------|
| Statut: | Fermé | Début: | 09 janvier 2018 |
| Priorité: | Normal | Echéance: | |
| Assigné à: | Frédéric Péters | % réalisé: | 0% |
| Catégorie: | | Temps estimé: | 0:00 heure |
| Version cible: | | Planning: | |
| Patch proposed: | Oui | | |

Description

À regarder les performances dans [#17645](#) il y a quelques zones facilement optimisables,

- le `extend_with_parent_cells` pourrait récupérer toutes les cellules potentielles en un coup, plutôt que faire ça parent par parent,
- la vérification pour les badges n'a pas besoin de récupérer les cellules pour de vrai, juste savoir qu'il y en a. (ça n'a pas des masses d'incidence mais dans ma conversion en [#17645](#) je suis passé par là, je le garde donc, et m'en sert pour bâtir le suivant),
- la modification par les cellules du contexte de "sœurs" devrait uniquement faire des requêtes SQL sur les cellules qui vraiment peuvent altérer le contexte. (sur une page ce n'est pas tellement important parce qu'on a déjà récupéré les cellules, par contre pour l'affichage ajax d'une cellule seule, c'est un vrai gain).

J'ai poussé ça en vrac dans `wip/performances`; je reviendrai dessus.

Révisions associées

Révision c8305b32 - 15 février 2018 09:44 - Frédéric Péters

misc: refactor `get_all_cell_types` to support a filtering function (#21052)

Révision e82f701f - 15 février 2018 09:44 - Frédéric Péters

perfs: refactor `get_badges` (#21052)

Révision a049d749 - 15 février 2018 09:44 - Frédéric Péters

perfs: optimize `extend_with_parent_cells` (#21052)

Révision 85b3ddc6 - 15 février 2018 09:44 - Frédéric Péters

perfs: limit cells considered for `cell.modify_global_context` (#21052)

Historique

#1 - 14 janvier 2018 09:43 - Frédéric Péters

- Fichier `0001-perfs-optimize-extend_with_parent_cells-21052.patch` ajouté
- Fichier `0002-perfs-refactor-get_badges-21052.patch` ajouté
- Fichier `0003-perfs-limit-cells-considered-for-cell.modify_global_.patch` ajouté
- Patch *proposed* changé de *Non* à *Oui*

Dans la branche, mais qui gagne son propre ticket, j'ai aussi fait [#21167](#).

Ça m'irait bien maintenant que les trois commits marqués [#21052](#) dans la branche <https://git.entrouvert.org/combo.git/log/?h=wip/performances> soient relus. (je les attache à ce ticket)

#2 - 17 janvier 2018 16:49 - Benjamin Dauvergne

Frédéric Péters a écrit :

Dans la branche, mais qui gagne son propre ticket, j'ai aussi fait [#21167](#).

Ça m'irait bien maintenant que les trois commits marqués [#21052](#) dans la branche <https://git.entrouvert.org/combo.git/log/?h=wip/performances> soient relus. (je les attache à ce ticket)

- 0001: il reste un truc un peu lent à cause de `Page.get_parents_and_self()` qui continue à faire du pointer chasing mais c'est à voir un autre jour en changeant la façon de gérer l'arbre. Pour le reste ça m'a l'air bon, je paraphrase le code pour voir si j'ai bien compris:
 - on récupère la liste des pages parents avant de résoudre les `ParentContentCell`,
 - ensuite on récupère d'un coup les cellules de toutes les pages parents + la page index,
 - on sépare les cellules de la page parent de la page en cours dans une variable `cells`
 - on remonte la chaîne des pages parentes pour chacune:
 - on parcourt `cells` si on trouve un `ParentContentCell` on remplace par les cellules de la page parent en cours d'analyse.

Je verrai comme optimisation en plus d'arrêter le parcours des pages parents dès qu'on ne trouve plus de `ParentContentCell` dans `cells`, avec un `else: break` dans la boucle `for` interne, et le code serait aussi peut-être plus clair avec:

```
for cell in cells:
    if isinstance(cell, ParentContentCell):
        # replace by parent page cells
        cells[i:i + 1] = cells_by_page[page_id]
        break
else: # no more ParentContentCell, stop following the parent page chain
    break
```

Aussi je remarque que c'est un peu gênant d'avoir `CellBase.get_cells()` et `ParentContentCell.get_cells()` nommées pareil et qui n'ont en fait aucun rapport. Je renommerai bien la deuxième en `get_parent_cells()` pour éviter les incompréhensions. Aussi le `hierarchy` qui est passé un peu partout est pas terrible, je me dis qu'on pourrait avoir une optimisation qui fait que `Page.get_parents_and_self()` met en cache la hiérarchie sur un premier appel, ça évite de passer la valeur un peu partout tout ça pour éviter de refaire ces requêtes, ainsi on pourra refaire sans souci le `get_parents_and_self()` dans `ParentContentCell.get_parent_cells()` sans se soucier de savoir si ça a déjà été appelé plus haut mais peut-être faut-il changer aussi les `CellBase.get_cells(page_id=selected_page.id)` en `CellBase.get_cells(page=selected_page)` pour être sûr que les cellules partagent le même objet `Page` (à vérifier quand même mais je crois que Django fait cette optimisation), on évite ainsi de modifier toutes les signatures pour y passer `hierarchy`. Et tant qu'à optimiser le nombre de requêtes il faudrait un test sur ce point.

- 0002: Le `catch` de `DoesNotExist` sur `Page(slug='index')` et le reste du code dans `ParentContentCell.get_cells()` me semble appartenir au patch précédent. Pour le reste je séparerai le patch en 2, les améliorations à l'API de `CellBase` puis l'application au problème de trouver les cellules qui ont une méthode `get_badge()`. Aussi peut-être que ce serait moins impactant d'ajouter un `class_filter` à `CellBase.get_cells()` ça évite de recopier le parcours de classe partout (`for klass in CellBase.get_cell_classes(...): cells.extend(...)`) qui jusqu'à présent était bien encapsulé dans `CellBase.get_cells()`. Je ne vois pas le rapport des modifications dans les tests de lingo, faut en faire un patch à part et si ça a un rapport dire lequel.
- 0003: et donc ici avec mes suggestions le code pourrait devenir simplement

```
other_cells = CellBase.get_cells(page_id=cell.page_id, class_filter=lambda k: bool(k.modify_global_context),
                                cell_filter=lambda c: c.is_visible(user=request.user))
```

, ça supprime 4 lignes. À voir si on gagne vraiment à ne pas résoudre `cell.page` en utilisant `cell.page_id` (j'ai l'impression qu'on va forcément faire une requête pour la page à un moment, un `select_related('page')` aux bons endroits suffirait à ce qu'on ne se pose plus la question).

#3 - 18 janvier 2018 09:42 - Frédéric Péters

- 0001 , <https://git.entrouvert.org/combo.git/commit/?h=wip/performances&id=a485cf8c72e2dd4ff8355bc2528910169a52dee1>

Je verrai comme optimisation en plus d'arrêter le parcours des pages parents [...]

Yep.

Je renommerai bien la deuxième en `get_parent_cells()` pour éviter les incompréhensions.

J'ai mis `get_parents_cells()` (`parentS`) mais oui, tout à fait plus clair ainsi.

ça évite de passer la valeur un peu partout [...] on évite ainsi de modifier toutes les signatures pour y passer `hierarchy`.

Ça reste quand même limité à deux/trois appels, je préfère rester explicite ici qu'ajouter la mise en cache.

Et tant qu'à optimiser le nombre de requêtes il faudrait un test sur ce point.

Yep, ajouté. Une requête SQL en plus par niveau dans la hiérarchie.

- 0002, splitté en <https://git.entrouvert.org/combo.git/commit/?h=wip/performances&id=ecf47358aaee80c0047c0431068f37a0d98b4e3> & <https://git.entrouvert.org/combo.git/commit/?h=wip/performances&id=8c4fa059fc2fa36701cc20c737f5c07fb28816ad>

Oui, j'ai déplacé correctement ce qui relevait du 0001 dans le précédent. Et les tests lingo je n'ai plus le souvenir précis mais déplacés dans leur propre commit. Et limité un premier commit à "refactor `get_all_cell_types` to support a filtering function" sans toucher aux badges.

Par contre je n'essaie pas d'unifier le parcours des classes, il reste quelques mini différences dont je n'évalue pas l'impact. (dans un cas on prête attention au flag "visible" qu'on utilise pour ne pas présenter dans l'UI du `/manage/` des cellules dépréciées, mais dans le cas du rendu de la page, on

passer quand même dessus parce qu'on veut effectivement conserver le rendu de la cellule si elle était déjà là) (et c'est parce que je ne suis pas du tout sûr que ça soit réellement ça la raison, je préfère ne pas toucher).

- 0003, <https://git.entrouvert.org/combo.git/commit/?h=wip/performances&id=7d142556de21c958016d90cb0ab08cc442b3e989>

Au vu de mon commentaire précédent, je le laisse en l'état.

#4 - 14 février 2018 17:51 - Anonyme

- Fichier `benchmark_public.py` ajouté

Voilà un prototype de tests de performance d'une des vues touchées par les patches ("menu_badges")

Il ne reste qu'à faire la même avec les autres vues touchées par la modif :

- skeleton
- page
- style
- error404

Et comparer les performances entre le master et le wip.

On peut utiliser soit la fonction de génération d'un arbre de pages orienté profondeur, soit d'autres données de test (pas de fixtures, donc je ne vois pas comment faire)

#5 - 14 février 2018 18:03 - Anonyme

Ack

#7 - 15 février 2018 09:47 - Frédéric Péters

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

Ma relecture du code ne me fait penser à rien sinon peut-être que tu comptes faire qui est de squasher les 2 premiers commits comme ils modifient chacun `get_cells()` par exemple.

La granularité a un peu été adaptée après la première relecture de Benjamin, je la laisse ainsi et pousse la série de commits.

```
commit 85b3ddc69702f625faa5583b7447763da169af8e
Author: Frédéric Péters <fpeters@entrouvert.com>
Date: Tue Jan 9 09:58:02 2018 +0100
```

```
perfs: limit cells considered for cell.modify_global_context (#21052)
```

```
commit a049d749e6a4ef7c55c136e6ae8d0588edd717e5
Author: Frédéric Péters <fpeters@entrouvert.com>
Date: Thu Jan 18 08:47:42 2018 +0100
```

```
perfs: optimize extend_with_parent_cells (#21052)
```

```
commit e82f701ffa1e60777a9af00d346890c319535e9b
Author: Frédéric Péters <fpeters@entrouvert.com>
Date: Thu Jan 18 08:50:21 2018 +0100
```

```
perfs: refactor get_badges (#21052)
```

```
commit 9d1f38e645191e5b7057f4b6d257eb40e0ecf150
Author: Frédéric Péters <fpeters@entrouvert.com>
Date: Thu Jan 18 08:48:11 2018 +0100
```

```
tests: use more precise checks for lingo cells
```

```
commit c8305b3251f32201d5c3bdce824ff60431dc098b
Author: Frédéric Péters <fpeters@entrouvert.com>
Date: Thu Jan 18 08:46:40 2018 +0100
```

```
misc: refactor get_all_cell_types to support a filtering function (#21052)
```

#8 - 15 février 2018 12:56 - Frédéric Péters

J'écrivais :

Il faut vraiment inclure uniquement la réponse dans les commentaires redmine, autrement ça fait un truc tout à fait illisible. (tu peux éditer les commentaires pour corriger ça)

Et je l'ai fait moi-même.

#9 - 15 février 2018 14:02 - Anonyme (Notes privées)

Je ne comprends pas de quoi tu parles, du coup, j'ai simplement édité mon commentaire pour y mettre simplement Ack

#11 - 23 décembre 2018 15:13 - Frédéric Péters

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

Fichiers

| | | | |
|---|---------|-----------------|-----------------|
| 0001-perfs-optimize-extend_with_parent_cells-21052.patch | 3,43 ko | 14 janvier 2018 | Frédéric Péters |
| 0002-perfs-refactor-get_badges-21052.patch | 5,37 ko | 14 janvier 2018 | Frédéric Péters |
| 0003-perfs-limit-cells-considered-for-cell.modify_global_.patch | 2,69 ko | 14 janvier 2018 | Frédéric Péters |
| benchmark_public.py | 2,8 ko | 14 février 2018 | Anonyme |