

w.c.s. - Development #65511

sur une recherche full text, en l'absence de critère de tri, trier par pertinence des résultats

20 mai 2022 12:05 - Frédéric Péters

Statut:	Fermé	Début:	20 mai 2022
Priorité:	Normal	Echéance:	
Assigné à:	Frédéric Péters	% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	Non
Patch proposed:	Oui		
Description			
Typiquement ça permettra à combo qui appelle /api/cards/whatever/list?q=toto d'obtenir les résultats classés "correctement".			
Demandes liées:			
Lié à w.c.s. - Development #16253: Trier par pertinence les résultats d'une r...		Fermé	10 mai 2017

Révisions associées

Révision a71e70c7 - 31 mai 2022 17:29 - Frédéric Péters

tests: add basic backoffice form fts search test (#65511)

Révision 97af09ca - 31 mai 2022 17:29 - Frédéric Péters

api/backoffice: order by rank (#65511)

Révision d97880ee - 31 mai 2022 17:29 - Frédéric Péters

misc: use a single sql query to get item ids (#65511)

Révision 10e74275 - 31 mai 2022 17:29 - Frédéric Péters

misc: merge filter and ordering queries (#65511)

Historique

#2 - 20 mai 2022 14:18 - Frédéric Péters

Du contexte, sur un appel /api/cards/whatever/list?q=toto

```
SELECT id FROM carddata_31_whatever WHERE fts @@ plainto_tsquery('toto')
```

d'où on récupère la liste des id, puis on fera un SELECT ... WHERE id IN (1,3,4,5,6).

En présence d'un order_by, on ajoute entre ces deux étapes une requête

```
SELECT id FROM carddata_31_whatever ORDER BY le_tri
```

(où on voit qu'on a perdu le critère de recherche full text, j'ai créé [#65525](#)).

L'idée serait, quand order_by n'est pas précisé, et qu'il y a une recherche fts, qu'on ait un tri par pertinence.

Ce qu'on aurait en django d'un façon type qs = Whatever.objects.annotate(rank=SearchRank(vector, query)).order_by('-rank').

Ce travail en deux/trois temps (récup des id, tri des id, récup des données) ne peut pas être réduit à une seule étape (à cause d'une compatibilité ancien stockage pickle), mais on pourrait réduire ça à (récup des id triés, récup des données).

#3 - 20 mai 2022 15:19 - Frédéric Péters

mais on pourrait réduire ça à (récup des id triés, récup des données).

C'était mon idée dans [#65525](#) mais non.

Mais je viens de taper un début de branche pour ici, qui fait

```
+ elif not anonymise and get_publisher().is_using_postgresql() and query:
```

```
+ criterias = [FtsMatch(query)] + (criterias or [])
+ ordered_ids = formdata_class.get_sorted_ids(order_by, clause=criterias)
```

et l'idée serait ici de mettre `order_by = 'rank'` et que `get_sorted_ids` se débrouille.

#4 - 25 mai 2022 12:22 - Frédéric Péters

- Assigné à mis à Pierre Ducroquet

Je m'occuperai de la partie code parce qu'il y a pas mal à démêler là-edans mais je veux bien les infos sur les requêtes qui seraient à faire pour un tri par pertinence.

#6 - 27 mai 2022 15:47 - Pierre Ducroquet

Niveau SQL y'a rien de bien sorcier, le critère d'order by est le suivant :

```
ORDER BY ts_rank(fts, to_tsquery('french', 'choucroute')) DESC
```

Ce qui donne en requête complète:

```
select id, fts from formdata_42_fete_de_la_saucisse where fts @@ to_tsquery('french', 'choucroute') order by t
s_rank(fts, to_tsquery('french', 'choucroute')) desc;
```

#7 - 27 mai 2022 15:49 - Frédéric Péters

- Assigné à changé de Pierre Ducroquet à Frédéric Péters

top, merci.

#8 - 30 mai 2022 09:57 - Frédéric Péters

- Fichier 0004-misc-merge-filter-and-ordering-queries-65511.patch ajouté
- Fichier 0003-misc-use-a-single-sql-query-to-get-item-ids-65511.patch ajouté
- Fichier 0002-api-backoffice-order-by-rank-65511.patch ajouté
- Fichier 0001-tests-add-basic-backoffice-form-fts-search-test-65511.patch ajouté
- Statut changé de Nouveau à Solution proposée
- Patch proposed changé de Non à Oui

- 0001 c'est quelques tests basiques posés avant de développer,
- 0002 c'est la réalisation minimale, construction d'un ORDER BY ts_rank (ce ticket pourrait s'arrêter ici),
- 0003 c'est l'idée de réduire les requêtes nécessaires pour la récupération des id, en spécialisant totalement `get_listing_item_ids()` en cas de stockage postgresql, en passant tous les critères dans une seule requête,
- 0004 c'est la suite, réduire encore d'une requête, en incluant également le tri dans la requête de récupération des id.

#9 - 31 mai 2022 16:37 - Lauréline Guérin

- Statut changé de Solution proposée à Solution validée

#10 - 31 mai 2022 17:29 - Frédéric Péters

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

```
commit 10e74275d62046d3d2ecba91779b5b3294844daf
Author: Frédéric Péters <fpeters@entrouvert.com>
Date: Sun May 29 12:58:18 2022 +0200
```

```
misc: merge filter and ordering queries (#65511)
```

```
commit d97880ee7256071141277acd055b7a1cb2bf829b
Author: Frédéric Péters <fpeters@entrouvert.com>
Date: Sun May 29 12:31:12 2022 +0200
```

```
misc: use a single sql query to get item ids (#65511)
```

```
commit 97af09ca2dcb20e0e199ab6307e2a3b467ee9af2
Author: Frédéric Péters <fpeters@entrouvert.com>
Date: Fri May 20 15:15:26 2022 +0200
```

api/backoffice: order by rank (#65511)

commit a71e70c79366918e2c0091b57033f72b18c4af5c
Author: Frédéric Péters <fpeters@entrouvert.com>
Date: Fri May 20 14:35:27 2022 +0200

tests: add basic backoffice form fts search test (#65511)

#11 - 31 mai 2022 19:14 - Transition automatique

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

#12 - 31 juillet 2022 04:42 - Transition automatique

Automatic expiration

#13 - 21 mai 2023 16:54 - Frédéric Péters

- Lié à Development #16253: Trier par pertinence les résultats d'une recherche full text ajouté

Fichiers

0004-misc-merge-filter-and-ordering-queries-65511.patch	3,72 ko	30 mai 2022	Frédéric Péters
0003-misc-use-a-single-sql-query-to-get-item-ids-65511.patch	8,68 ko	30 mai 2022	Frédéric Péters
0002-api-backoffice-order-by-rank-65511.patch	4,4 ko	30 mai 2022	Frédéric Péters
0001-tests-add-basic-backoffice-form-fts-search-test-65511.patch	2,62 ko	30 mai 2022	Frédéric Péters