

Passerelle - Development #39563

avoir la recherche dans les logs chercher également dans le contenu des messages

05 février 2020 14:19 - Frédéric Péters

Statut:	Fermé	Début:	05 février 2020
Priorité:	Normal	Echéance:	
Assigné à:	Valentin Deniaud	% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	Non
Patch proposed:	Oui		
Description			
Sur un problème comme #39562 j'aimerais pouvoir prendre un bout de ce qui est affiché, ex: BASTION, et chercher ça dans les logs.			

Révisions associées

Révision 823b366c - 17 février 2020 16:13 - Valentin Deniaud

views: make log searching more exhaustive (#39563)

Historique

#1 - 11 février 2020 18:09 - Valentin Deniaud

- Assigné à mis à Valentin Deniaud

Pas aussi trivial qu'il n'y paraît. Je pense :

1. restreindre aux cas où l'endpoint parle au format wcs, avec un len() pour vérifier que 'data' ne contient pas 3000 éléments
2. regarder seulement l'attribut 'text' des résultats
3. créer un nouveau modèle avec pour unique but de stocker un de ces attributs, et une ForeignKey vers l'object ResourceLog associé¹

Ce qui au final devrait permettre un lookup simple du type `resource_logs.filter(text_result_set__text__icontains=query)`.

¹ et ça me donne l'impression d'en faire trop. Face à ça, pour s'éviter le modèle en plus, il y a la solution qui consiste à faire pas assez, genre ne stocker que le premier résultat, ou faire mal, genre les concaténer.

#2 - 11 février 2020 18:36 - Frédéric Péters

restreindre aux cas où l'endpoint parle au format wcs

Souvent je veux chercher dans les appels (ou les réponses) vers l'extérieur, en fait dans le vrac qu'est "extra"; en SQL

```
select * from base_resourcelog where extra::text ilike '%query%';
```

(caster le jsonb en texte)

Et à lire django-jsonfield, class `JSONFieldIContainsLookup(ContainsLookupMixin, IContains):`.

Du coup, `.filter(extra__contains=query)` doit marcher de manière générale.

(oui il y aura parfois du déchet parce que la recherche va se faire sur les clés et les valeurs, plutôt qu'uniquement les valeurs, mais je pense que ça vaut l'affaire quand même).

#3 - 12 février 2020 16:41 - Valentin Deniaud

- Fichier `0001-views-extend-log-search-to-responses-39779.patch` ajouté

- Statut changé de Nouveau à Solution proposée

- Patch proposed changé de Non à Oui

Frédéric Péters a écrit :

Du coup, `.filter(extra__contains=query)` doit marcher de manière générale.

Non, cf la [doc](#), contains n'est pas le même que pour un TextField et fait un match exact.

(caster le jsonb en texte)

Ça par contre ça marche du tonnerre, merci !

Pour la suite je suis allé de problème en problème, ce code est tellement utilisé que la moindre modif casse des choses inattendues. Le but étant simplement de mettre la réponse dans extra.

Je me suis aussi aperçu que la réponse était déjà stockée dans le cas d'une erreur, donc je m'en suis inspiré (sérialisation puis tronquage des données).

Le seul problème qu'il reste c'est mon try/finally qui change l'ordre des lignes de log, la requête au connecteur va apparaître après la requête au logiciel métier. Si c'est mal, il faut ajouter un champ uuid à ResourceLog, et ajouter la réponse à posteriori si la requête n'a pas levé d'exception.

#4 - 12 février 2020 17:00 - Frédéric Péters

J'imaginai ici qu'on chercherait juste dans l'existant, rien ajouter; i.e. juste la modif à GenericViewLogsConnectorView.

#5 - 12 février 2020 17:02 - Frédéric Péters

Ensuite, on doit déjà logger le nécessaire ou presque dans la ligne existante, l'étape utile derrière étant de faire le lien entre les différents logs produits par un unique appel : [#38157](#).

#6 - 12 février 2020 17:13 - Valentin Deniaud

- Fichier 0001-views-make-log-searching-more-exhaustive-39563.patch ajouté

OK, c'est juste que ça ne marchera pas pour l'exemple du ticket.

#7 - 12 février 2020 17:30 - Frédéric Péters

Pourtant,

```
passerelle=> select id from base_resource_log where extra::text ilike '%bastion%' limit 10;
 id
-----
 25219
 25221
 25223
 25225
 25231
 25241
 25246
 25251
 25254
 26168
(10 rows)
```

#8 - 12 février 2020 18:06 - Valentin Deniaud

Moi tu trouves les requêtes entre passerelle et le connecteur métier, pas la réponse du connecteur. La question est de savoir si on dit que ça marche par chance ou qu'au contraire ça marche comme ça 99% du temps.

Il y a un certain nombre de contre-exemples, les connecteurs simples genre csv/json (cf le test dans le premier patch) ou streets/api geo qui ont les données en local. Dans ces cas là, la recherche sur la réponse ne fonctionnera pas.

#9 - 15 février 2020 10:12 - Frédéric Péters

En fait, je ne souhaitais pas retirer message__icontains=query, mais bien faire une recherche qui couvre également extra, Q(message__icontains=query) | Q(text_extra__icontains=query)).

#10 - 17 février 2020 15:13 - Valentin Deniaud

- Fichier 0001-views-make-log-searching-more-exhaustive-39563.patch ajouté

Je ne l'ai pas fait parce que tout ce qui est dans message est aussi mis dans extra (sauf peut-être la méthode HTTP dans certains endroits, mais on pourrait aussi l'ajouter). Après j'entends qu'on veuille anticiper des évolutions futures de message.

#11 - 17 février 2020 15:36 - Frédéric Péters

Non, ok, c'est sans doute je n'arrive pas encore très bien à capter les différences de compréhension, qui t'amènent à faire la première version du patch puis ce commentaire "tu trouves les requêtes entre passerelle et le connecteur métier, pas la réponse du connecteur".

Si message__icontains n'apporte rien de plus, inutile de le mettre, on peut très bien se dire que tout se trouvera toujours bien dans extra.

Et sans doute qu'il y a clairement un truc à clarifier dans ce qui est loggué et comment c'est loggué, et que tu parlais dans cette direction. Pour le coup je regarde les tickets et il y a [#31774](#) qui correspond peut-être ?

#12 - 17 février 2020 16:28 - Valentin Deniaud

- Fichier `0001-views-make-log-searching-more-exhaustive-39563.patch` ajouté

Effectivement j'étais en train de faire [#31774](#), je vais aller parler là bas. Pour ici, ce patch suffit, donc.

#13 - 17 février 2020 16:33 - Frédéric Péters

- Statut changé de *Solution proposée* à *Solution validée*

Go.

#14 - 17 février 2020 17:43 - Valentin Deniaud

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

```
commit 823b366ce236df24feb520404ed0cb0d40d1d914
Author: Valentin Deniaud <vdeniaud@entrouvert.com>
Date: Wed Feb 12 17:03:01 2020 +0100
```

```
views: make log searching more exhaustive (#39563)
```

#15 - 19 février 2020 18:15 - Frédéric Péters

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

Fichiers

<code>0001-views-extend-log-search-to-responses-39779.patch</code>	6,55 ko	12 février 2020	Valentin Deniaud
<code>0001-views-make-log-searching-more-exhaustive-39563.patch</code>	1,43 ko	12 février 2020	Valentin Deniaud
<code>0001-views-make-log-searching-more-exhaustive-39563.patch</code>	1,47 ko	17 février 2020	Valentin Deniaud
<code>0001-views-make-log-searching-more-exhaustive-39563.patch</code>	1,43 ko	17 février 2020	Valentin Deniaud