

## Passerelle - Bug #54442

### opendatasoft: trier les résultats

31 mai 2021 16:50 - Nicolas Roche

<b>Statut:</b>	Fermé	<b>Début:</b>	31 mai 2021
<b>Priorité:</b>	Normal	<b>Echéance:</b>	
<b>Assigné à:</b>	Nicolas Roche	<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>	Oui		
<b>Description</b>			
Pour que les résultats soient triés lors de leur présentation sous forme de liste.			
<b>Demandes liées:</b>			
Lié à Passerelle - Bug #55698:.opendatasoft: permettre de configurer le param...			Fermé 20 juillet 2021

### Révisions associées

#### Révision 27380a6d - 27 août 2021 14:12 - Nicolas Roche

opendatasoft: add sort field (#54442)

### Historique

#### #1 - 11 juin 2021 12:37 - Nicolas Roche

- Fichier 0001-opendatasoft-order-search-results-by-text-54442.patch ajouté
- Tracker changé de Support à Bug
- Statut changé de Nouveau à Solution proposée
- Assigné à mis à Nicolas Roche
- Patch proposed changé de Non à Oui

#### #3 - 11 juin 2021 14:11 - Nicolas Roche

- Sujet changé de.opendatasoft: trier les résultats à.opendatasoft: trier les résultats

#### #5 - 25 juin 2021 14:25 - Frédéric Péters

```
result.sort(key=itemgetter('text'))
```

Perso davantage habitué à `key=lambda x: x.get('text')` qui ne demande pas à chercher d'où vient ce que fait `itemgetter`.

..

Mais sur le fond ça ne pense pas nécessairement approprié d'imposer ce tri, parce que j'imagine que par défaut le tri fourni par opendatasoft peut ou doit avoir un sens.

#### #6 - 25 juin 2021 16:30 - Nicolas Roche

- Fichier 0001-opendatasoft-order-search-results-by-text-54442.patch ajouté
- Sujet changé de.opendatasoft: trier les résultats à.opendatasoft: trier les résultats<https://documentation-resources.opendatasoft.com/api/datasets/1.0/search?exclude.publisher=GeoNam>Je pensais que c'était une bonne chose de trier d'office sur 'text' pour que les listes dans wcs, les champs&sort=-modified

(c'était pour que les listes affichées dans wcs, les champs soient triés).

Rien à voir, mais j'étais passé à côté, sort fait parti des paramètres acceptés par l'API

<https://help.opendatasoft.com/apis/ods-search-v1/#dataset-search-api>

Sorts results by the specified field. By default, the sort is descending. A minus sign - may be used to perform an ascending sort. Sorting is only available on numeric fields (Integer, Decimal), date fields (Date, DateTime), and on text fields that have the sortable annotation

ex: <https://documentation-resources.opendatasoft.com/api/datasets/1.0/search?exclude.publisher=GeoNames&sort=-modified>

**#7 - 25 juin 2021 16:51 - Frédéric Péters**

(c'était pour que les listes affichées dans wcs, les champs soient triés).

Ce qui n'a pas systématiquement du sens.

Je serais pour ajouter à la définition d'une requête un champ permettant de taper l'attribut sur lequel faire le tri (juste un champ texte, pas chercher à produire une liste), que cela soit passé lors de l'appel à opendatasoft, et qu'en l'absence, il ne soit rien fait concernant le tri.

**#8 - 25 juin 2021 16:58 - Frédéric Péters**

- *Sujet changé de opendatasoft: trier les résultats*  
<https://documentation-resources.opendatasoft.com/api/datasets/1.0/search?exclude.publisher=GeoNam> Je pensais que c'était une bonne chose de trier d'office sur 'text' pour que les listes dans wcs, les champs&sort=modified à opendatasoft: trier les résultats

**#9 - 25 juin 2021 19:43 - Nicolas Roche**

ajouter à la définition d'une requête un champ permettant de taper l'attribut sur lequel faire le tri

Fait,  
mais tout bien réfléchi, je me dis que l'on pourrait juste autoriser le champs dans l'URL et le propager, sans le stocker dans la définition des requêtes, comme c'est fait par exemple pour le paramètre 'limit'.

**#10 - 25 juin 2021 19:44 - Nicolas Roche**

- *Fichier 0001-opendatasoft-add-sort-field-54442.patch ajouté*

**#11 - 25 juin 2021 20:05 - Nicolas Roche**

- *Fichier 0001-opendatasoft-add-sort-field-54442.patch ajouté*

Avec le paramètre sort également passé comme paramètre au endpoint search .

**#12 - 28 juin 2021 10:15 - Nicolas Roche**

- *Statut changé de Solution proposée à En cours*

juste autoriser le champs dans l'URL et le propager

Si ça convient, je repars là dessus.

**#13 - 19 juillet 2021 15:09 - Nicolas Roche**

- *Fichier 0001-opendatasoft-manage-sort-parameter-54442.patch ajouté*

- *Statut changé de En cours à Solution proposée*

**#14 - 19 juillet 2021 15:51 - Thomas Noël**

Un peu perdu : on ne peut plus passer le sort dans une requête pré-configurée ?

**#15 - 19 juillet 2021 16:33 - Nicolas Roche**

Non, pas plus que pour le paramètre "limit" où l'on fait juste passe-plat.

Dis-moi si tu préfères <https://dev.entrouvert.org/issues/54442#note-11> et à ce moment là j'intégrerais également le paramètre "limit" dans le paramétrage de la requête.

**#16 - 19 juillet 2021 16:39 - Thomas Noël**

Je lisais juste la note #7 :

Je serais pour ajouter à la définition d'une requête un champ permettant de taper l'attribut sur lequel faire le tri (juste un champ texte, pas chercher à produire une liste), que cela soit passé lors de l'appel à opendatasoft, et qu'en l'absence, il ne soit rien fait concernant le tri.

et je vois que ce n'est pas ce qui est proposé dans ton patch.

#### #17 - 19 juillet 2021 19:42 - Nicolas Roche

Oui, j'ai proposé le patch allant dans ce sens <https://dev.entrouvert.org/issues/54442#note-11> puis me suis ravisé <https://dev.entrouvert.org/issues/54442#note-12>. Comme il n'y a pas eu de réponse depuis, je propose <https://dev.entrouvert.org/issues/54442#note-13>.

#### #18 - 20 juillet 2021 10:41 - Thomas Noël

Ce sont des choses que tu as décidé tout seul... sans dire pourquoi tu as changé d'avis. Pour ma part, je reste sur la demande de Frédéric, plus cohérente. On va pas s'amuser à rajouter des ?order= abscons dans les appels aux requêtes.

#### #19 - 20 juillet 2021 11:58 - Nicolas Roche

- Lié à Bug #55698: *opendatasoft: permettre de configurer le paramètre limit dans les requêtes ajouté*

#### #20 - 20 juillet 2021 12:12 - Nicolas Roche

- Fichier 0001-opendatasoft-add-sort-field-54442.patch ajouté

Voilà.

#### #21 - 20 juillet 2021 14:01 - Thomas Noël

Je n'ai pas testé, mais habituellement quand on interroge une API de type "search", elle renvoie les résultats dans l'ordre de pertinence. Ca serait dommage de perdre ça, et même gênant dans le cas de l'utilisation de l'autocomplétion. Peux-tu vérifier si les résultats remontent par ordre de pertinence avec de l'opendatasoft qu'on a à disposition ?

Si lors d'une recherche les résultats sont bien renvoyés par ordre de pertinence, alors ça voudra dire qu'il ne faut pas appliquer le "sort" sur les recherches, mais uniquement lors de la remonté de la liste exhaustive (q is None).

Du moins c'est mon avis...

#### #22 - 20 juillet 2021 18:41 - Nicolas Roche

Oui, je le répète (et je vois que c'était pas super clair), ici on fait juste passe plat, il n'y a aucun traitement dans le connecteur concernant le trie. (Au début j'ai effectivement proposé un patch pour appliquer le trie sur le gabarit de text, puis Fred m'a indiqué comme toi que c'est à opendatasoft de procéder au trie en amont.)

#### #23 - 21 juillet 2021 00:14 - Thomas Noël

Nicolas Roche a écrit :

Oui, je le répète (et je vois que c'était pas super clair), ici on fait juste passe plat, il n'y a aucun traitement dans le connecteur concernant le trie. (Au début j'ai effectivement proposé un patch pour appliquer le trie sur le gabarit de text, puis Fred m'a indiqué comme toi que c'est à opendatasoft de procéder au trie en amont.)

Oui, mais quand on envoie un q=xxx et un sort=yyy à l'API search d'opendatasoft, il se passe quoi ? C'est ça que j'aimerais que tu vérifies : que le sort=yyy envoyé ne modifie par l'ordre d'une recherche, pour laquelle on s'attend à recevoir les résultats dans l'ordre de pertinence. Si c'est bien le cas alors ton patch est ok, sinon il faudra retirer le sort= quand un q= est envoyé.

#### #24 - 28 juillet 2021 12:06 - Nicolas Roche

Alors voyons si je te suis et prenons l'exemple des merveilles du monde : <https://examples.opendatasoft.com/api/records/1.0/search/?dataset=world-heritage-unesco-list&q=merveille>

Le Mont-Saint-Michel et sa baie sont plus pertinents que les pyramides de Guizeh en tant que "merveille".

```
$ curl 'https://examples.opendatasoft.com/api/records/1.0/search/?dataset=world-heritage-unesco-list&q=merveille' | json_pp | grep name_fr
(2° position) "name_fr" : "Mont-Saint-Michel et sa baie",
(3)           "name_fr" : "Memphis et sa nécropole - les zones des pyramides de Guizeh à Dahchour",
```

(jusqu'ici je suis d'accord)

Si je trie sur la date d'inscription au patrimoine mondial, vu qu'ils y ont été inscrits la même année, alors j'aimerais retrouver cette même préférence régionale.

```
$ curl 'https://examples.opendatasoft.com/api/records/1.0/search/?dataset=world-heritage-unesco-list&q=merveille&sort=date_inscribed' | json_pp | grep name_fr
(8)           "name_fr" : "Memphis et sa nécropole - les zones des pyramides de Guizeh à Dahchour",
```

```
(9)      "name_fr" : "Mont-Saint-Michel et sa baie",
```

Mince, je tri à l'envers pour voir...

```
$ curl 'https://examples.opendatasoft.com/api/records/1.0/search/?dataset=world-heritage-unesco-list&q=merveil
le&sort=-date_inscribed' | json_pp | grep name_fr
(1)      "name_fr" : "Memphis et sa nécropole - les zones des pyramides de Guizeh à Dahchour",
(2)      "name_fr" : "Mont-Saint-Michel et sa baie",
```

Re-mince, visiblement le sort=yyy envoyé modifie l'ordre d'une recherche, pour laquelle on s'attend à recevoir les résultats dans l'ordre de pertinence. Reste que j'ai peut-être compris de travers la notion d'ordre de pertinence.

quand on envoie un q=xxx et un sort=yyy

Si je l'applique au contexte où les deux paramètres q= et sort= sont renseignés alors ça semble tenir.

```
$ curl 'https://examples.opendatasoft.com/api/records/1.0/search/?dataset=world-heritage-unesco-list&q=mer&sort=-date_inscribed' | json_pp | grep name_fr
(2)      "name_fr" : "Memphis et sa nécropole - les zones des pyramides de Guizeh à Dahchour",
(5)      "name_fr" : "Mont-Saint-Michel et sa baie",
```

Est-ce que je creuse ce second point, ou le premier point répond à ta question et je retire le sort= quand un q= est envoyé ?

#### #25 - 28 juillet 2021 12:31 - Frédéric Péters

Tu te prends beaucoup trop la tête à inventer des situations et à faire intervenir des données qui ne sont pas visibles, etc. c'est impossible à suivre.

Je ne sais pas ce qui amène ce ticket, les seuls tickets liés sont postérieurs.

Est-ce que je creuse ce second point

Je dirais que non non non faut surtout arrêter de creuser.

Tu prends une pièce de monnaie tu la lances et si c'est pile tu retires sort= quand q= est envoyé, face tu laisses.

#### #26 - 28 juillet 2021 16:16 - Nicolas Roche

- Fichier 0003-opendatasoft-do-not-send-sort-parameter-with-q-54442.patch ajouté

- Fichier 0002-opendatasoft-do-not-send-empty-q-parameter-54442.patch ajouté

- Fichier 0001-opendatasoft-add-sort-field-54442.patch ajouté

Pile.

0002 : J'en profite pour retirer l'envoi du paramètre "q=" quand il n'est pas renseigné (ça faisait moche dans les tests et ça fonctionne tout aussi bien).

0003 : Je retire également l'envoi du paramètre "sort=" sur la recherche par identifiant. Comme on attend un seul enregistrement, à mon avis ça n'a pas non plus de sens de demander un trie.

#### #27 - 28 juillet 2021 16:38 - Thomas Noël

0003: le « if sort and not (q or id): » peut agréablement devenir un « elif sort is not None: » posé après le « if id is not None: ... elif q is not None: ... » (c-à-d qu'on ne gère sort qu'en l'absence de id et q)

Au passage, mettre sort=None dans la signature, comme pour id et q.

Aussi, les «update» pour modifier une seule clé d'un dictionnaire, c'est très moche. Je trouverais plus joli de faire « params['sort'] = sort »

Enfin, pourquoi faire 3 patches ? Un seul ça serait sans doute plus facile à lire/comprendre, je pense.

#### #28 - 28 juillet 2021 17:15 - Nicolas Roche

- Fichier 0001-opendatasoft-add-sort-field-54442.patch ajouté

Merci.

Enfin, pourquoi faire 3 patches ?

Je pensais ainsi faciliter la relecture et augmenter mes chance pour que ce patch soit validé.

C'était aussi dans l'idée de rebasculer un jour sur le tri dans la recherche (la motivation première de ce ticket) :  
une liste en autocomplétion sur les nom de rues dans wcs renvoie des choix dans un ordre (celui d'opendatasoft) étrange.  
Mais j'abandonne cette idée parce que le trie opendatasoft n'est pas systématique sur les champs texte.

**#29 - 29 juillet 2021 01:12 - Thomas Noël**

- Statut changé de Solution proposée à Solution validée

**#30 - 27 août 2021 14:19 - Nicolas Roche**

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

```
commit 27380a6dd11ae7a0c2a5647e5fe879d477765bda
Author: Nicolas ROCHE <nroche@entrouvert.com>
Date: Fri Jun 25 19:24:46 2021 +0200
```

```
opendatasoft: add sort field (#54442)
```

**#31 - 30 août 2021 12:17 - Frédéric Péters**

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

**Fichiers**

0001-opendatasoft-order-search-results-by-text-54442.patch	5,01 ko	11 juin 2021	Nicolas Roche
0001-opendatasoft-order-search-results-by-text-54442.patch	4,24 ko	25 juin 2021	Nicolas Roche
0001-opendatasoft-add-sort-field-54442.patch	6,67 ko	25 juin 2021	Nicolas Roche
0001-opendatasoft-add-sort-field-54442.patch	9,34 ko	25 juin 2021	Nicolas Roche
0001-opendatasoft-manage-sort-parameter-54442.patch	6,51 ko	19 juillet 2021	Nicolas Roche
0001-opendatasoft-add-sort-field-54442.patch	9,34 ko	20 juillet 2021	Nicolas Roche
0002-opendatasoft-do-not-send-empty-q-parameter-54442.patch	6,46 ko	28 juillet 2021	Nicolas Roche
0003-opendatasoft-do-not-send-sort-parameter-with-q-54442.patch	4,04 ko	28 juillet 2021	Nicolas Roche
0001-opendatasoft-add-sort-field-54442.patch	9,35 ko	28 juillet 2021	Nicolas Roche
0001-opendatasoft-add-sort-field-54442.patch	13,6 ko	28 juillet 2021	Nicolas Roche