

## Combo - Development #61083

### dataviz, extension à l'API pour permettre des sous-filtres

25 janvier 2022 17:10 - Valentin Deniaud

<b>Statut:</b>	Fermé	<b>Début:</b>	25 janvier 2022
<b>Priorité:</b>	Normal	<b>Echéance:</b>	
<b>Assigné à:</b>	Valentin Deniaud	<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		

#### Description

Dans l'API stat des différentes briques, on expose d'abord un catalogue des statistiques disponibles avec d'éventuels filtres par statistiques, type :

```
{
  'data': [
    {
      'name': 'Nombre de demandes',
      'url': 'https://wcs.dev.publik.love/api/statistics/forms/count/',
      'id': 'forms_counts',
      'filters': [
        {
          'id': 'form',
          'label': 'Formulaire',
          'options': [
            { 'id': '_all', 'label': 'Tous' },
            { 'id': '1', 'label': 'Inscription piscine' },
            { 'id': '2', 'label': 'Commande de bacs' },
          ],
          'required': True,
          'default': '_all',
        },
      ],
    },
  ],
}
```

Dans cet exemple wcs expose une statistique « Nombre de demandes », filtrable selon le formulaire.

Imaginons maintenant que l'on veuille permettre un filtrage selon les champs du formulaire, à la bijou. Cela dépend entièrement du choix réalisé, on ne les connaît pas à l'avance.

Normalement si je sélectionne un formulaire ça donne :

```
GET https://wcs.dev.publik.love/api/statistics/forms/count/?form=2
```

```
{
  'data': {
    'series': [{ 'data': [1, 2], 'label': 'Nombre de demandes' }],
    'x_labels': ['2020-07', '2020-08']
  },
  'err': 0
}
```

Et je me dis que ça pourrait renvoyer des sous-filtres dans une clé supplémentaire :

```
GET https://wcs.dev.publik.love/api/statistics/forms/count/?form=2
```

```
{
```

```
'data': {
  'series': [{'data': [1, 2], 'label': 'Nombre de demandes'}],
  'x_labels': ['2020-07', '2020-08'],
  'subfilters':
    [
      {
        'id': 'nb_bacs',
        'label': 'Nombre de bacs',
        'options': [
          {'id': '1', 'label': 'Un'},
          {'id': '2', 'label': 'Deux'},
        ],
      },
    ],
  },
  'err': 0
}
```

ce qui permet ensuite de réinterroger avec ?nb\_bacs=xxx.

## Révisions associées

### Révision 30d1483f - 15 février 2022 17:01 - Valentin Deniaud

dataviz: add support for subfilters (#61083)

## Historique

### #1 - 25 janvier 2022 17:35 - Valentin Deniaud

- Fichier 0001-wip.patch ajouté
- Statut changé de Nouveau à Solution proposée
- Patch proposed changé de Non à Oui

Patch tout simple pour illustrer.

Le problème avec ça c'est qu'il y a besoin de deux clics sur le bouton pour avoir les nouveaux champs :

- Sélection du formulaire
- Clic sur « Enregistrer »
  - Les champs sont rafraîchis
  - Le graphe avec les nouvelles données et les sous-filtres correspondant au formulaire choisi est récupéré en ajax
  - Donc les sous-filtres n'apparaissent pas dans les champs de la cellule
- Nouveau clic sur « Enregistrer »
  - Les champs sont rafraîchis, et les sous-filtres sont là

Je ne sais pas encore comment faire, sûrement du JS pour forcer le rafraîchissement du graphe avant celui du formulaire.

### #2 - 25 janvier 2022 17:36 - Valentin Deniaud

- Statut changé de Solution proposée à En cours

### #3 - 07 février 2022 18:04 - Valentin Deniaud

- Fichier 0001-dataviz-add-support-for-subfilters-61083.patch ajouté
- Statut changé de En cours à Solution proposée

Même pas de javascript, à la place une petite astuce pour recharger le formulaire après le save() (lire PageEditCellView.form\_valid pour comprendre).

Aussi par rapport à la description, j'ajoute une clé has\_subfilters au filtre qui déclenche l'apparition des sous-filtres, pour ne pas recharger les données inutilement.

### #4 - 15 février 2022 15:04 - Frédéric Péters

Je verrais bien un commentaire au-dessus de :

```
self.__init__()
```

(ou découper l'init pour avoir une méthode avec un nom explicite qui serait appelée ici).

#### #5 - 15 février 2022 17:11 - Valentin Deniaud

- Fichier 0001-dataviz-add-support-for-subfilters-61083.patch ajouté

Frédéric Péters a écrit :

Je verrais bien un commentaire au-dessus de :

[...]

(ou découper l'init pour avoir une méthode avec un nom explicite qui serait appelée ici).

Yep, il s'agit de 1/ virer les champs obsolètes 2/ ajouter les nouveaux, je découpe pour le 2/ mais le 1/ était géré par le super().\_\_init\_\_(), donc il faut faire un peu autrement (et c'est tant mieux).

#### #6 - 15 février 2022 18:46 - Frédéric Péters

- Statut changé de Solution proposée à Solution validée

#### #7 - 16 février 2022 10:05 - Valentin Deniaud

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

```
commit 30d1483f03d17344bb9c7adc5f6d3d4e5e7ce1b7
Author: Valentin Deniaud <vdeniaud@entrouvert.com>
Date: Mon Jan 24 18:13:24 2022 +0100
```

```
dataviz: add support for subfilters (#61083)
```

#### #8 - 17 février 2022 20:17 - Transition automatique

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

#### #9 - 24 avril 2022 04:42 - Transition automatique

Automatic expiration

### Fichiers

0001-wip.patch	3,54 ko	25 janvier 2022	Valentin Deniaud
0001-dataviz-add-support-for-subfilters-61083.patch	9,39 ko	07 février 2022	Valentin Deniaud
0001-dataviz-add-support-for-subfilters-61083.patch	10,9 ko	15 février 2022	Valentin Deniaud