

## Combo - Development #48865

### dataviz: récupérer les statistiques depuis d'autres briques

26 novembre 2020 11:19 - Valentin Deniaud

<b>Statut:</b>	Fermé	<b>Début:</b>	26 novembre 2020
<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		
<b>Description</b>			
Avec comme visée d'afficher ce qu'expose authentic avec <a href="#">#48845</a> .			
<b>Demandes liées:</b>			
Lié à Combo - Development #49173: dataviz: mettre le catalogue des visualisat...		<b>Fermé</b>	<b>07 décembre 2020</b>

#### Révisions associées

##### Révision 4f878ac9 - 22 décembre 2020 11:38 - Valentin Deniaud

dataviz: split get\_chart into several methods (#48865)

##### Révision b615b7fa - 22 décembre 2020 11:38 - Valentin Deniaud

tests: add dataviz null visualization (#48865)

##### Révision 18fdd8a8 - 22 décembre 2020 11:38 - Valentin Deniaud

dataviz: handle new api to get statistics from elsewhere (#48865)

##### Révision d8fbda7d - 22 décembre 2020 11:38 - Valentin Deniaud

dataviz: avoid crash if no table data (#48865)

#### Historique

##### #1 - 01 décembre 2020 17:01 - Valentin Deniaud

- Fichier 0001-tests-decrease-indentation-level-in-dataviz-48865.patch ajouté
- Fichier 0003-dataviz-save-available-visualizations-in-db-48865.patch ajouté
- Fichier 0002-tests-format-dataviz-using-black-48865.patch ajouté
- Fichier 0004-dataviz-handle-new-api-to-get-statistics-from-elsewh.patch ajouté
- Statut changé de Nouveau à Solution proposée
- Patch proposed changé de Non à Oui

Version wip, me reste à écrire des tests et prendre en charge les filtres déclarés dans le retour de l'API.

Pour me faciliter la vie, j'ai eu envie de commencer par simplifier les tests :

- 0001, réduire les niveaux d'indentation en utilisant des fixtures
- 0002, quitte à pourrir l'historique, autant en profiter pour passer un coup de black

Ensuite, 0003 est un gros préliminaire : actuellement la liste des visualisations est récupérée à chaque affichage du formulaire + au moment du save. Impossible de fonctionner comme ça lorsqu'il n'y a n sites à aller visiter. Donc transition vers un système de cache des statistiques dispos, mis à jour toutes les heures. Je trouve que ça simplifie le code, quand même quelques subtilités à noter :

- Data migration pour mettre à jour les cellules existantes. Elle est splittée en 3 fichiers parce que postgres ne tolère pas qu'on change les données et le schéma en même temps (en vrai la migration en un seul fichier passe chez moi en local, mais pas dans le test, dunno why).
- En prévision du patch d'après, les visualisation sont désormais systématiquement préfixées du nom du site. Ça ne va pas rendre génial puisque bijoe s'appelle par défaut « Statistiques », je suis preneur d'idées pour faire mieux.

Et le ticket commence à être traité seulement dans 0004 : un nouveau setting (à poser par hobo ?) indique dans quelles briques aller chercher les stats. Puis prise en charge du nouveau format de l'API, c'est deux lignes mais le diff est un peu gros car je bouge du code en introduisant une séparation entre configuration de graphe, commun à tout le monde, et parsing de la réponse, séparé entre bijoe et les autres.

## #2 - 01 décembre 2020 17:34 - Valentin Deniaud

Valentin Deniaud a écrit :

prendre en charge les filtres déclarés dans le retour de l'API.

Ah et aussi, avant même ça authentic requiert qu'on lui dise quel regroupement temporel utiliser, ça passe par un paramètre `?time_interval={day,month,year}`. Ce qui se traduit par un nouveau champ à choix dans la cellule, tranquille, mais son affichage est problématique, je vois deux solutions pas idéales : l'afficher par défaut et le masquer quand on sait que la visualisation vient de bijoe, au premier save (ce que fait le patch actuellement), ou inversement ne l'afficher qu'après, mais ça requiert un rafraîchissement manuel de la page... Je me dis que ce cas s'est peut-être déjà présenté pour d'autres cellules et qu'il y a une manière de gérer ça ?

## #3 - 01 décembre 2020 17:54 - Frédéric Péters

un nouveau setting (à poser par hobo ?)

Je serais pour un setting qui soit la liste des URL, plutôt que des identifiants de brique, l'idée étant qu'ainsi on pourra facilement ajouter des statistiques ad hoc, peut-être aussi simplement qu'en posant deux fichiers json sur un serveur statique.

mais ça requiert un rafraîchissement manuel de la page

Normalement pas, après un clic sur "enregistrer" le formulaire est réaffiché tel qu'envoyé par le serveur (pour les messages d'erreur mais il peut gagner un champ, je pense que ça doit marcher). C'est sans doute plus facile de faire ça que de jouer au javascript pour dynamiquement afficher telle ou telle option.

Par contre, `?time_interval=...` je ne vois pas ce qui dans l'API listant les statistiques indique que ça existe, et je suis un peu frileux à partir là-dessus dès maintenant, j'aurais préféré que soit listées explicitement comme statistiques "... par mois", "... par année", etc.

Mon idée là-dessus c'est qu'on va très régulièrement vouloir ça et que ça m'irait de demander aux briques de pouvoir simplement faire une agrégation par jour, et la main à combo pour grouper ensuite en semaines/mois/années, plutôt qu'avoir ce travail répété partout. (cela en me disant que côté volume de données, l'agrégation par jour fournit quelque chose dans des volumes qui se traitent rapidement via Python, qu'on peut ignorer l'optimisation possible par la requête SQL).

## #4 - 02 décembre 2020 14:20 - Valentin Deniaud

- *Statut changé de Solution proposée à En cours*

Frédéric Péters a écrit :

Je serais pour un setting qui soit la liste des URL, plutôt que des identifiants de brique, l'idée étant qu'ainsi on pourra facilement ajouter des statistiques ad hoc, peut-être aussi simplement qu'en posant deux fichiers json sur un serveur statique.

Les identifiants de brique ça a le mérite de bien marcher en cas d'import/export entre recette et prod, peut-être ajouter le support des urls en plus et pas à la place ?

Par contre, `?time_interval=...` je ne vois pas ce qui dans l'API listant les statistiques indique que ça existe, et je suis un peu frileux à partir là-dessus dès maintenant, j'aurais préféré que soit listées explicitement comme statistiques "... par mois", "... par année", etc.

Authentic expose déjà 6 visualisations différentes, j'aimerais éviter d'en avoir 18. Dans ton pad tu indiques que les regroupements temporels c'est 75% des stats, je me disais qu'il y avait moyen qu'un champ dédié à ça soit toujours utile.

Mais je suis d'accord que c'est prématuré, et qu'il n'y a aucune raison que ça ne soit pas un filtrage optionnel comme celui sur ou/service, déclaré comme tel dans l'API de listing, je vais plutôt faire comme ça.

## #5 - 02 décembre 2020 16:14 - Frédéric Péters

Les identifiants de brique ça a le mérite de bien marcher en cas d'import/export entre recette et prod, peut-être ajouter le support des urls en plus et pas à la place ?

Yep en effet.

## #6 - 07 décembre 2020 16:40 - Valentin Deniaud

- *Fichier 0001-dataviz-split-get\_chart-into-several-methods-48865.patch ajouté*

- *Fichier 0003-dataviz-avoid-crash-if-no-table-data-48865.patch ajouté*

- Fichier 0002-dataviz-handle-new-api-to-get-statistics-from-elsewh.patch ajouté

- Statut changé de En cours à Solution proposée

Et donc décision de splitter tout ça dans plusieurs tickets, les histoires de db dont je parlais dans mes premiers patches c'est maintenant [#49173](#) (prérequis à ce ticket tout de même), et les filtres ce sera [#49175](#).

ajouter le support des urls en plus

Décision aussi de faire ça dans un prochain ticket.

Ce qui permet de se concentrer sur ce qui est demandé ici :

- 0001 parce que la méthode get\_chart de bijoe est bien trop longue, et qu'il faut permettre la réutilisation de certains bouts.
- 0002 le gros du boulot.
- 0003 correction en passant d'un bug déjà là, render\_table de pygal crashe si pas de données.

#### #7 - 14 décembre 2020 12:00 - Valentin Deniaud

- Lié à Development #49173: dataviz: mettre le catalogue des visualisations dispo en db ajouté

#### #8 - 14 décembre 2020 16:07 - Frédéric Péters

Pour info, j'ai obtenu à un moment

```
File "/home/fred/src/eo/combo/combo/apps/dataviz/views.py", line 43, in dataviz_graph
    chart = cell.get_chart(
File "/home/fred/src/eo/combo/combo/apps/dataviz/models.py", line 246, in get_chart
    data = self.process_one_dimensional_data(chart, data)
File "/home/fred/src/eo/combo/combo/apps/dataviz/models.py", line 347, in process_one_dimensional_data
    data = self.hide_values(chart, data)
File "/home/fred/src/eo/combo/combo/apps/dataviz/models.py", line 356, in hide_values
    x_labels, new_data = zip(*[(label, value) for label, value in zip(chart.x_labels, data) if value])
ValueError: not enough values to unpack (expected 2, got 0)
```

(mais sans savoir ce qui avait été reçu à ce moment)

#### #9 - 16 décembre 2020 10:37 - Valentin Deniaud

Frédéric Péters a écrit :

Pour info, j'ai obtenu à un moment

[...]

(mais sans savoir ce qui avait été reçu à ce moment)

Merci, j'ai pu reproduire et je corrige.

#### #10 - 22 décembre 2020 09:08 - Frédéric Péters

- Statut changé de Solution proposée à Solution validée

Go.

#### #11 - 22 décembre 2020 11:39 - Valentin Deniaud

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

```
commit d8fbda7d4241db8e001a4dbf5abba44885c5bd0d
Author: Valentin Deniaud <vdeniaud@entrouvert.com>
Date: Mon Dec 7 12:35:56 2020 +0100
```

```
dataviz: avoid crash if no table data (#48865)
```

```
commit 18fdd8a8c4f2f8f5e69bf4ca88f29a16bd2ae4a2
Author: Valentin Deniaud <vdeniaud@entrouvert.com>
Date: Mon Nov 30 14:10:39 2020 +0100
```

```
dataviz: handle new api to get statistics from elsewhere (#48865)
```

```
commit b615b7fa998b907b079c2b76b22692cfcffdbe49
```

Author: Valentin Deniaud <vdeniaud@entrouvert.com>  
Date: Wed Dec 16 11:09:55 2020 +0100

tests: add dataviz null visualization (#48865)

commit 4f878ac96ae31e42c735720d6d49843e40788a1f  
Author: Valentin Deniaud <vdeniaud@entrouvert.com>  
Date: Thu Dec 3 15:08:17 2020 +0100

dataviz: split get\_chart into several methods (#48865)

## #12 - 22 décembre 2020 15:16 - Frédéric Péters

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

### Fichiers

---

0001-tests-decrease-indentation-level-in-dataviz-48865.patch	38,9 ko	01 décembre 2020	Valentin Deniaud
0003-dataviz-save-available-visualizations-in-db-48865.patch	34,1 ko	01 décembre 2020	Valentin Deniaud
0002-tests-format-dataviz-using-black-48865.patch	6,5 ko	01 décembre 2020	Valentin Deniaud
0004-dataviz-handle-new-api-to-get-statistics-from-elsewh.patch	14,6 ko	01 décembre 2020	Valentin Deniaud
0001-dataviz-split-get_chart-into-several-methods-48865.patch	7,55 ko	07 décembre 2020	Valentin Deniaud
0003-dataviz-avoid-crash-if-no-table-data-48865.patch	2,1 ko	07 décembre 2020	Valentin Deniaud
0002-dataviz-handle-new-api-to-get-statistics-from-elsewh.patch	18 ko	07 décembre 2020	Valentin Deniaud