

## Authentic 2 - Development #47467

### Produire des statistiques simples

08 octobre 2020 11:31 - Benjamin Dauvergne

<b>Statut:</b>	Fermé	<b>Début:</b>	08 octobre 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		

#### Description

On part vers la production d'une API interne (pour exploitation soit en BO soit en API, on va pas traiter ça pour l'instant).

Les filtrages :

- Période
- Découpage : aucun, par année, par mois, par semaine

Pour chaque élément du découpage :

- Nombre de login total et par type
- Nombre de login par ou et par type (pourcentage pour l'OU pour chaque type: mot de passe / FC)
- Nombre de login par service et par type (pourcentage pour chaque service : mot de passe / FC)
- Nombre d'enregistrement total
- Nombre d'enregistrement par ou et par type (pourcentage pour l'OU pour chaque type: mot de passe / FC)
- Nombre d'enregistrement par service et par type (pourcentage pour chaque service : mot de passe / FC)

## Connexions

	Janvier	Février	...
mot de passe	100 (66 %)	50 (50 %)	...
FC	50 (34 %)	50 (50 %)	...

## Par collectivité

	Janvier		Février		...
	mot de passe	FC	mot de passe	FranceConnect	...
Collectivité par défaut	100 (66 %)	50 (34 %)	50 (50 %)	50 (50 %)	...

## Par service

	Janvier		Février		...
	mot de passe	FC	mot de passe	FranceConnect	...
Portail usager	100 (66 %)	50 (34 %)	50 (50 %)	50 (50 %)	...

## Créations de compte

	Janvier	Février	...
--	---------	---------	-----

mot de passe	100 (66 %)	50 (50 %)	...
FC	50 (34 %)	50 (50 %)	...

## Par collectivité

	Janvier		Février		...
	mot de passe	FC	mot de passe	FranceConnect	...
Collectivité par défaut	100 (66 %)	50 (34 %)	50 (50 %)	50 (50 %)	...

## Par service

	Janvier		Février		...
	mot de passe	FC	mot de passe	FranceConnect	...
Portail usager	100 (66 %)	50 (34 %)	50 (50 %)	50 (50 %)	...

### Demandes liées:

Bloqué par Authentic 2 - Development #47155: Avoir sur les objets un journal...

**Fermé**

**30 septembre 2020**

### Révisions associées

#### Révision c1345a33 - 01 décembre 2020 17:36 - Valentin Deniaud

journal: add event type statistics (#47467)

### Historique

#### #1 - 08 octobre 2020 11:37 - Benjamin Dauvergne

- Bloqué par Development #47155: Avoir sur les objets un journal des modifications et sur les utilisateurs, en plus, un journal des actions ajouté

#### #2 - 09 novembre 2020 10:56 - Benjamin Dauvergne

- Sujet changé de Produire une page de statistiques simples à Produire -une page de- des statistiques simples

- Description mis à jour

#### #3 - 09 novembre 2020 10:56 - Benjamin Dauvergne

- Sujet changé de Produire -une page de- des statistiques simples à Produire des statistiques simples

#### #4 - 18 novembre 2020 17:28 - Valentin Deniaud

- Fichier 0001-wip.patch ajouté

- Statut changé de Nouveau à Solution proposée

- Assigné à mis à Valentin Deniaud

- Patch proposed changé de Non à Oui

Benjamin Dauvergne a écrit :

On part vers la production d'une API interne (pour exploitation soit en BO soit en API, on va pas traiter ça pour l'instant).

Un peu trop théorique pour que j'arrive à bien avancer. J'ai écrit une méthode qui est capable de renvoyer les stats attendues par la description, je veux bien des pointeurs vers où aller ensuite.

Ça s'utilise en faisant genre

```
>>> from authentic2.journal_event_types import UserLogin
>>> UserLogin.get_statistics(group_by_time='month', group_by_field='how')
<EventQuerySet [{'month': datetime.datetime(2020, 10, 1, 0, 0, tzinfo=<DstTzInfo 'Europe/Paris' CEST+2:00:00 DST>), 'how': 'password-on-https', 'count': 15}, {'month': datetime.datetime(2020, 11, 1, 0, 0, tzinfo=<DstTzInfo 'Europe/Paris' CET+1:00:00 STD>), 'how': 'fc', 'count': 1}, {'month': datetime.datetime(2020, 11, 1, 0, 0, tzinfo=<DstTzInfo 'Europe/Paris' CET+1:00:00 STD>), 'how': 'password-on-https', 'count': 11}]>
```

## #6 - 18 novembre 2020 18:42 - Benjamin Dauvergne

Valentin Deniaud a écrit :

Benjamin Dauvergne a écrit :

On part vers la production d'une API interne (pour exploitation soit en BO soit en API, on va pas traiter ça pour l'instant).

Un peu trop théorique pour que j'arrive à bien avancer. J'ai écrit une méthode qui est capable de renvoyer les stats attendues par la description, je veux bien des pointeurs vers où aller ensuite.

Tu réponds en partie à la demande, il faudrait à partir de ton code produire les données pour les cas listés: connexion par mode de connexion, connexion par services et modes de connexion, idem pour création de compte. Ce serait bien si ça sortait directement les bons labels pour les modes de connexion qu'on ait pas à maintenir ça ailleurs. Le cas par collectivité veut dire par collectivité du service, c'est spécifique à GLC, mais c'est important, c'est le plus gros demandeur de ces stats.

## #7 - 19 novembre 2020 18:29 - Valentin Deniaud

- Fichier 0001-journal-add-event-type-statistics-47467.patch ajouté

Voilà, je laisse de côté pour le moment les pourcentages (pygal ou autre sait potentiellement faire, donc à voir plus tard si il y en a vraiment besoin) et l'idée de formater les dates de la manière la moins verbeuse possible (amélioration pour après).

Pour les stats sur la méthode seule je formate les données comme suggéré par Fred, et pour les stats à plusieurs dimensions (méthode + service/ou), j'invente un truc dont je ne sais pas ce qu'il vaut (cf test).

## #8 - 19 novembre 2020 18:45 - Frédéric Péters

et pour les stats à plusieurs dimensions (méthode + service/ou), j'invente un truc dont je ne sais pas ce qu'il vaut (cf test).

Je ne vois pas dans les tests la situation, que je ne suis pas sûr de comprendre.

Dans le pad, j'ai ajouté tout à l'heure une section sur l'API pour lister les visualisations disponibles (pour exposer ces séries de statistiques).

Côté x\_labels ils doivent pour moi correspondre à ce qui sera affiché à l'utilisateur, i.e. pour l'année, "2020" et pas "2020-01-01T00:00:00+01:00". (ou alors on part tout de suite sur des métadonnées supplémentaires pour décrire le x\_labels, genre "x\_datatype": "time:years").

Truc aussi, pour ces événements qui sont liés au temps, il ne faudrait pas que l'absence de données sur une période fasse sauter celle-ci, concrètement si 5 personnes de logguent en février, 0 en mars, 10 en avril, je ne voudrais pas que mars soit zappé des données. (ou alors on part pareil à se dire qu'il y aura métadonnées et que combo saura que c'est une série chronologique et aura en responsabilité de compléter, ça me va).

## #9 - 19 novembre 2020 21:39 - Benjamin Dauvergne

Frédéric Péters a écrit :

Truc aussi, pour ces événements qui sont liés au temps, il ne faudrait pas que l'absence de données sur une période fasse sauter celle-ci, concrètement si 5 personnes de logguent en février, 0 en mars, 10 en avril, je ne voudrais pas que mars soit zappé des données. (ou alors on part pareil à se dire qu'il y aura métadonnées et que combo saura que c'est une série chronologique et aura en responsabilité de compléter, ça me va).

Autant c'est simple pour l'axe temporel X autant ce sera vraiment plus compliqué pour combo pour les Y même si ça a moins de chance d'arriver. « aucun événement sur la période pour le service X par exemple » ce serait dommage de voir le service disparaître des statistiques dans ce cas, même s'il n'y a rien à en dire, c'est perturbant. Dans bijoe je m'assure d'avoir soit la totalité des valeurs dans la requête (pour l'axe temporel) soit je génère l'axe hors requête et je complète au moment de générer les résultats.

Je ne suis néanmoins pas gêné que le problème ne soit pas traité à ce stade si on en a conscience.

## #10 - 23 novembre 2020 18:18 - Valentin Deniaud

- Fichier 0001-journal-add-event-type-statistics-47467.patch ajouté

Nouvelle itération, avec du code plus clair et des tests qui devraient passer.

Frédéric Péters a écrit :

et pour les stats à plusieurs dimensions (méthode + service/ou), j'invente un truc dont je ne sais pas ce qu'il vaut (cf test).

Je ne vois pas dans les tests la situation, que je ne suis pas sûr de comprendre.

La situation c'est un peu genre les loop\_labels que peut renvoyer bijoe j'ai l'impression, ça correspond au deuxième tableau de benj dans la description par exemple. Les tests étaient foireux, maintenant ça devrait mieux se voir, genre

```
'series': {
  'Second OU': [
    {'label': 'FranceConnect', 'data': [2, None]},
    {'label': 'password', 'data': [2, 1]},
  ],
  'Default organizational unit': [{'label': 'password', 'data': [None, 2]}],
}
```

Côté x\_labels ils doivent pour moi correspondre à ce qui sera affiché à l'utilisateur, i.e. pour l'année, "2020" et pas "2020-01-01T00:00:00+01:00". (ou alors on part tout de suite sur des métadonnées supplémentaires pour décrire le x\_labels, genre "x\_datatype": "time:years").

Truc aussi, pour ces événements qui sont liés au temps, il ne faudrait pas que l'absence de données sur une période fasse sauter celle-ci, concrètement si 5 personnes se loguent en février, 0 en mars, 10 en avril, je ne voudrais pas que mars soit zappé des données. (ou alors on part pareil à se dire qu'il y aura des métadonnées et que le combo saura que c'est une série chronologique et aura en responsabilité de compléter, ça me va).

C'est vrai que j'avais supposé 2) direct mais qu'en fait ajouter 1) ici ne coûterait pas grand chose. Après on peut aussi poser le problème en terme de duplication de code, c'est peut-être mieux si le code qui complète vit dans le combo et pas dans les n briques qui vont être amenées à produire des stats.

En fait, troisième voie, on voit qu'on pourrait bien simplifier le code ici en n'ayant pas ce besoin d'avoir un x\_labels commun à toutes les séries, pour l'exemple du dessus ça ferait

```
{
  "series": [
    {
      "label": "password-on-https",
      "x_labels": ['2020-10', '2020-11'],
      "data": [15, 11],
    },
    {
      "label": "fc",
      "x_labels": ['2020-10'],
      "data": [7],
    }
  ]
}
```

Et c'est aussi consommable très facilement par pygal, en utilisant ce qui est prévu pour les dates <http://www.pygal.org/en/stable/documentation/types/xy.html#date>.

Benjamin Dauvergne a écrit :

Je ne suis néanmoins pas gêné que le problème [pas de valeurs en Y] ne soit pas traité à ce stade si on en a conscience.

Yep, on verra plus tard.

### #11 - 23 novembre 2020 19:02 - Frédéric Péters

La situation c'est un peu genre les loop\_labels que peut renvoyer bijoe j'ai l'impression

Mais c'est un truc qui génère plusieurs graphes, qui peuvent avoir des libellés différents sur leurs axes (et du coup pas repris dans le combo qui affiche un seul graphe). Je n'ai pas vraiment l'impression que dans tes exemples tu tapes vraiment sur ce qui me pose problème, mais la description du ticket laisse ça discrètement hors des exemples.

	Janvier		Février		...
	mot de passe	FC	mot de passe	FranceConnect	...
Collectivité par défaut	100 (66 %)	50 (34 %)	50 (50 %)	50 (50 %)	...
Collectivité une autre	100 (66 %)	50 (34 %)	50 (50 %)	50 (50 %)	...
Collectivité encore une	100 (66 %)	50 (34 %)	50 (50 %)	50 (50 %)	...

Tu représenterais ça comment dans l'API, et avec quoi comme représentation dans un graphe unique, avec quel code pygal ?

Ma réponse à ça est d'ignorer la question et de laisser à l'agent qui configure le portail de configurer plusieurs graphes s'il le souhaite, parce que dans la liste des données possibles fournies par Authentic il y aura "méthodes de connexion, par mois", "méthodes de connexion, par mois (collectivité par défaut)", "méthodes de connexion, par mois (une autre collectivité)", etc.

**#12 - 24 novembre 2020 00:20 - Benjamin Dauvergne**

Frédéric Péters a écrit :

La situation c'est un peu genre les loop\_labels que peut renvoyer bijoe j'ai l'impression

Tu représenterais ça comment dans l'API, et avec quoi comme représentation dans un graphe unique, avec quel code pygal ?

Ce genre de tableau n'est pas représentable en graphique, par contre pour une seule collectivité/un seul service on peut empiler les chiffres par mode d'authentification ou bien utiliser plusieurs lignes.

**#13 - 24 novembre 2020 10:14 - Valentin Deniaud**

Je récapitule avant de me lancer dans une nouvelle itération, il faut :

- oublier ce tableau, virer le code qui permet de le produire
- rajouter du code pour avoir les données des méthodes d'authentification pour un service/ou fixé
- rajouter du code pour avoir le nombre total d'authentification pour chaque service/ou, toutes méthodes confondues

Et plutôt partir sur des x\_labels affichables tels quels, ou sur l'idée qu'il y aura des métadonnées et que ce sera à combo de le faire ?

**#14 - 24 novembre 2020 10:39 - Frédéric Péters**

Et plutôt partir sur des x\_labels affichables tels quels, ou sur l'idée qu'il y aura des métadonnées et que ce sera à combo de le faire ?

À un moment passer la main à combo avec des données qualifiées, genre "x\_labels\_type": "time:month"; soit dès maintenant, soit plus tard (mais alors avoir des libellés directement intelligibles). (et je pense que 2020-11 même si pas idéal est intelligible, serait ok dès maintenant, même sans prise en charge d'un x\_labels\_type côté combo).

rajouter du code pour avoir les données des méthodes d'authentification pour un service/ou fixé

Dans l'immédiat oui, plus loin j'imaginerais que dans les métadonnées fournies par le graphe "nombre de connexions" il y ait une info genre

```
filters: [{"id": "ou", "label": "Collectivités", "option": [{"id": "", "label": "Toutes"}, {"id": "ou-1": "1", "label": "...}...}]}
```

Et le choix se ferait lors de la configuration de la cellule, sur base de ces filtres, plutôt que sur base d'une liste de données reprenant toutes les variations d'OU ou de services.

**#15 - 24 novembre 2020 15:55 - Valentin Deniaud**

- Fichier 0001-journal-add-event-type-statistics-47467.patch ajouté

Revoici, avec les bonnes stats, les libellés intelligibles et aussi un axe temporel sans trous.

Frédéric Péters a écrit :

plus loin j'imaginerais que dans les métadonnées fournies par le graphe "nombre de connexions"

Mais ça pour le coup ça ne concerne pas le morceau qu'on construit ici (dans la description, API interne). Pour un prochain ticket donc.

**#16 - 26 novembre 2020 08:16 - Benjamin Dauvergne**

- Statut changé de Solution proposée à Solution validée

Ok ça fait déjà une base.

**#17 - 01 décembre 2020 17:36 - Valentin Deniaud**

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

```
commit e64d09f119bdf98a45f899569b2508fb945f9c4b
Author: Valentin Deniaud <vdeniaud@entrouvert.com>
Date: Wed Nov 18 17:23:21 2020 +0100
```

```
journal: add event type statistics (#47467)
```

**#18 - 04 décembre 2020 09:11 - Frédéric Péters**

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

## Fichiers

---

0001-wip.patch	2,76 ko	18 novembre 2020	Valentin Deniaud
0001-journal-add-event-type-statistics-47467.patch	13,6 ko	19 novembre 2020	Valentin Deniaud
0001-journal-add-event-type-statistics-47467.patch	18,8 ko	23 novembre 2020	Valentin Deniaud
0001-journal-add-event-type-statistics-47467.patch	19,2 ko	24 novembre 2020	Valentin Deniaud