

## Passerelle - Development #16051

### connecteur d'association adresse → secteur

26 avril 2017 16:18 - Frédéric Péters

<b>Statut:</b>	Nouveau	<b>Début:</b>	26 avril 2017
<b>Priorité:</b>	Normal	<b>Echéance:</b>	
<b>Assigné à:</b>		<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>	Non		
<b>Description</b>			
<p>Le connecteur contrib/nancypoll permet à partir de nom et numéro de rue de trouver un bureau de vote, le connecteur externe des encombrants 3M permet entre autres de trouver à partir de nom de rue et commune le prestataire en charge, etc. Le connecteur contrib/nancypoll se trouve maintenant également utilisé pour tout autre chose ailleurs.</p> <p>Il me semble opportun de développer un connecteur générique assurant de la meilleure des manières ce boulot, plutôt que dupliquer/détourner.</p>			

### Historique

#### #1 - 26 avril 2017 16:27 - Serghei Mihai

Je vois aussi le besoin de pouvoir retourner ce "secteur" (ou autre chose) également à partir des coordonnées géographiques.

#### #2 - 26 avril 2017 16:32 - Frédéric Péters

Oui, mais le côté coordonnées géographiques, ça doit venir avec en backend un SIG gérant ça, alors qu'ici j'étais sur un système autonome.

#### #3 - 26 avril 2017 16:41 - Serghei Mihai

Tout à fait d'accord. Mais je pense à des cas comme Villejuif, qui n'a pas de SIG, mais juste un fichier CSV avec toutes les adresses et leur coordonnées géographiques pour déterminer le secteur.  
Mais bon, partons déjà ton idée initiale.

#### #4 - 26 avril 2017 16:45 - Frédéric Péters

Tout à fait d'accord. Mais je pense à des cas comme Villejuif, qui n'a pas de SIG, mais juste un fichier CSV avec toutes les adresses et leur coordonnées géographiques pour déterminer le secteur.

Oui, c'est l'objet de ce ticket, visiblement pas clair du tout.

#### #7 - 01 octobre 2018 19:37 - Frédéric Péters

Comparaison :

- street\_start\_number, présent Nancy & Arles,
- street\_end\_number, pareil,
- street\_side, I/P à Nancy, IMPAIR/PAIR à Arles,
- code, identifiant unique de rue à Arles, (n'existe pas dans le tableau Nancy)
- street\_name à Nancy, type/libelle à Arles (ex: type: RUE, libelle: DE L'ABBE GREGOIRE)
- id, code, text, address, canton, à Nancy, informations sur le résultat (ici id/code/libellé/adresse du bureau de vote, + canton); zone à Arles, bref, les données de l'objet recherché (bureau de vote, zone, etc.)

(dans les données de Nancy, non exploitées, il y a aussi complement\_debut/complement\_fin, dans la pratique on ne trouve pas de rue où un numéro est partagé entre deux zones selon le complément).

J'en serais presque à dire que ça doit être fait sous forme de filtre spécialisé dans le connecteur tableur, façon opérateur de comparaison particulier, genre address\_match(row, street=query.get('street'), number=query.get('number'), side=query.get('number')). (en imaginant que dans "row" on a le contenu de la ligne du tableau, je n'ai pas regardé exactement comment ça marchait).

#### #8 - 04 octobre 2018 09:52 - Serghei Mihai

Ça doit être possible déjà sans cet opérateur via une requête un peu plus complexe, genre:

`query.get('street') == street and query.get('number') <= end_number and query.get('number') => start_number`

#### #9 - 04 octobre 2018 12:29 - Frédéric Péters

Et gérer les pairs/impairs etc.

#### #10 - 09 octobre 2018 00:00 - Serghei Mihai

Modifier la colonne "parité" pour y mettre "0" pour pair et "1" pour impair, et dans la requête faire par exemple: `int(query.get('numero')) % 2 == int(numero)`

#### #11 - 10 octobre 2018 10:48 - Thomas Noël

Faire un connecteur générique avec un format de fichier qui ne soit pas tordu pour coller à une requête sur un CSV, ça simplifierait la vie des utilisateurs de ce connecteur, àmha.

#### #12 - 10 octobre 2018 14:07 - Serghei Mihai

Si le format de fichier est bien établi, genre code, voie, numero\_debut, numero\_fin, parite, secteur on aurait besoin de 2 requetes: une pour avoir l'aide à la saisie pour les noms des voies et l'autre pour calculer le secteur, à ajouter au connecteur CSV et c'est tout.  
Ou je rate quelque chose?

#### #13 - 10 octobre 2018 14:56 - Thomas Noël

Serghei Mihai a écrit :

Si le format de fichier est bien établi, genre code, voie, numero\_debut, numero\_fin, parite, secteur on aurait besoin de 2 requetes: une pour avoir l'aide à la saisie pour les noms des voies et l'autre pour calculer le secteur, à ajouter au connecteur CSV et c'est tout.  
Ou je rate quelque chose?

Par exemple chercher une rue avec la gestion des accents (on pourrait certes avoir un outil qui nous aide en général sur les CSV pour ça, mais bon).

Aussi, à côté du secteur, faudrait taper un id ou slug de secteur.

Et comment on fait pour les bis/ter/etc que les gens vont vouloir taper.

Et les rues dans numéros.

Bref tout ça fait une tonne de choses à gérer dans la requête CSV, donc de la config à faire à chaque fois, et un jour on veut faire évoluer le truc il faut le faire sur 50 clients, pas terrible.

Et comme je m'imagine comme toujours des montagnes, peut-être que ton idée est la bonne, il faut juste aller au bout et écrire les requêtes.

#### #14 - 10 octobre 2018 15:00 - Frédéric Péters

Je ne sais pas à quelle idée tu fais référence, mais parce qu'en effet gérer le détail avec uniquement les opérateurs de base, je n'y crois pas, je suggérerais l'ajout d'un "opérateur" `address_match()`. (mais c'était avant tout pour fournir une réponse rapide, alors que personne ne semblait vouloir partir sur un nouveau connecteur, qui pourrait être plus riche).

#### #15 - 21 juillet 2021 17:13 - Thomas Noël

- *Planning mis à Non*