

Passerelle - Support #26582

commande pour changer le fichier tableau d'une CSVDataSource

20 septembre 2018 12:20 - Benjamin Dauvergne

Statut:	Fermé	Début:	20 septembre 2018
Priorité:	Normal	Echéance:	
Assigné à:	Benjamin Dauvergne	% réalisé:	100%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	
Patch proposed:	Oui		
Description			
Pour répondre à la demande du ticket #26299, proposition: passerelle-manage csv-datasource change-csv <slug> blabla.csv			

Révisions associées

Révision f219ffce - 01 octobre 2018 16:53 - Benjamin Dauvergne

new command to change csv_file (fixes #26582)

Historique

#2 - 25 septembre 2018 11:14 - Benjamin Dauvergne

- Assigné à mis à Benjamin Dauvergne

#3 - 25 septembre 2018 17:09 - Benjamin Dauvergne

- Fichier 0001-manage-resource-from-command-line-fixes-26582.patch ajouté
- Statut changé de Nouveau à Solution proposée
- Patch proposed changé de Non à Oui

Je n'ai pas encore ajouté les tests mais je veux bien des retours sur la forme générale du truc. Je suis parti sur l'idée d'avoir une commande à tout faire sur les ressources.

```
$ passerelle-manage resource list-type --resource-type csvdatasource.csvdatasource
- title
  type: string
- description
  type: string
- log_level
  type: string
  choices:
    NOTSET: NOTSET
    DEBUG: DEBUG
    INFO: INFO
    WARNING: WARNING
    ERROR: ERROR
    CRITICAL: CRITICAL
- csv_file
  type: file
  help text: Supported file formats: csv, ods, xls, xlsx
- columns_keynames
  type: string
  help text: ex: id,text,data1,data2
- skip_header
  type: string
- sheet_name
  type: string
```

```
$ resource new csvdatasource.csvdatasource --string-field title coucou --string-field log_level WARNING --file
-field csv_file ~/tableau.csv --string-field description "blablabla"
```

```
$ ./manage.py resource list csvdatasource.csvdatasource
```

```
id          : 11
title       : coucou2
description : blablabla
slug        : coucou2
log_level   : WARNING
csv_file    : csv/.bashrc_Nc1M7EC
columns_keynames : id, text
skip_header : False
_dialect_options : {u'lineterminator': u'\r\n', u'skipinitialspace': False, u'quoting': 0, u'delimiter': u'
', u'quotechar': u'"', u'doublequote': False}
sheet_name  :
users       : []
```

```
$ resource update csvdatasource.csvdatasource coucou --file-field csv_file ~/tableau.csv
```

```
$ resource delete csvdatasource.csvdatasource coucou --file-field csv_file ~/tableau.csv
```

#4 - 25 septembre 2018 17:30 - Frédéric Péters

J'en serais resté à la demande, j'ai peur qu'on se galère sur des changements futurs à devoir garder en marche une commande à tout faire.

#5 - 25 septembre 2018 19:34 - Benjamin Dauvergne

- Fichier *0001-new-command-to-change-csv_file-fixes-26582.patch* ajouté

Voilà, beaucoup plus simple, aucun contrôle.

#6 - 01 octobre 2018 12:54 - Emmanuel Cazenave

Niveau système de fichier, à la lecture du code je ne comprends pas si les csv se retrouveront au même endroit que si ils étaient uploadés dans l'UI.

Peut-être si tu pouvais mettre cela en exergue dans un test ?

#7 - 01 octobre 2018 13:20 - Benjamin Dauvergne

Emmanuel Cazenave a écrit :

Niveau système de fichier, à la lecture du code je ne comprends pas si les csv se retrouveront au même endroit que si ils étaient uploadés dans l'UI.

C'est à dire au même endroit ? Le répertoire dépend du champ du modèle, chemin différera par contre, il va vouloir sauver le fichier avec le même nom, voir qu'un existe déjà, ajouter _1/2/3.etc.. et écrire.

Peut-être si tu pouvais mettre cela en exergue dans un test ?

Hmm.. je ne comprends pas.

#8 - 01 octobre 2018 13:46 - Emmanuel Cazenave

Quand tu upload un fichier csv dans l'UI à la création du connecteur, il finit dans `/var/lib/passerelle/tenants/nom-du-tenant/media/csv/`.

Ma question est : "est ce que ta commande assure bien ça ?"

#9 - 01 octobre 2018 15:17 - Benjamin Dauvergne

Emmanuel Cazenave a écrit :

Quand tu upload un fichier csv dans l'UI à la création du connecteur, il finit dans `/var/lib/passerelle/tenants/nom-du-tenant/media/csv/`.

À partir du moment où tu lances bien la commande via "tenant_command" oui, c'est le backend de storage issu de hobo.multitenant qui gère cela.

Ma question est : "est ce que ta commande assure bien ça ?"

Non elle ne s'en occupe pas, c'est automatique (je vois pas plus comment le code dans passerelle assure bien cela).

#10 - 01 octobre 2018 15:55 - Emmanuel Cazenave

- Statut changé de Solution proposée à Solution validée

Bon j'ai fait le test à la main et effectivement la totomagie marche.

#11 - 01 octobre 2018 16:55 - Benjamin Dauvergne

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

- % réalisé changé de 0 à 100

Appliqué par commit [f219ffceb9e081cc91965078d171d82decb9757d](https://commit-hash).

#12 - 03 décembre 2018 15:15 - Benjamin Dauvergne

- Statut changé de *Résolu* (à déployer) à *Fermé*

Fichiers

0001-manage-resource-from-command-line-fixes-26582.patch	9,36 ko 25 septembre 2018	Benjamin Dauvergne
0001-new-command-to-change-csv_file-fixes-26582.patch	4,41 ko 25 septembre 2018	Benjamin Dauvergne