

Authentic 2 - Development #31184

python3 : uniformiser le comportement de la méthode magique `__str__` des OU

06 mars 2019 22:12 - Paul Marillonnet

Statut:	Fermé	Début:	06 mars 2019
Priorité:	Normal	Echéance:	
Assigné à:	Paul Marillonnet	% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	Non
Patch proposed:	Oui		
Description			
Sans redéfinition de cette méthode, en python3, quelle que soit l'OU on a toujours :			
<pre>>>> whatever_ou 'OrganizationalUnit object'</pre>			
ce qui, en plus de casser les tests, ne donne pas beaucoup d'information sur l'OU.			
On doit redéfinir cette méthode pour imiter le comportement déjà présent en python2.			
Demandes liées:			
Lié à Authentic 2 - Development #28276: Fonctionner avec Python3 pour Django1.11		Fermé	23 novembre 2018

Révisions associées

Révision 43769fb2 - 02 mai 2019 10:52 - Paul Marillonnet

python3: py2 unicode compatible `str` magid methods (#31184)

Historique

#1 - 06 mars 2019 22:15 - Paul Marillonnet

- Fichier `0001-python3-ou.__str__-magid-method-uniformization-31184.patch` ajouté
- Statut changé de Nouveau à Solution proposée
- Patch proposed changé de Non à Oui

#2 - 06 mars 2019 22:29 - Paul Marillonnet

- Lié à Development #28276: Fonctionner avec Python3 pour Django1.11 ajouté

#3 - 25 mars 2019 10:56 - Benjamin Dauvergne

- Assigné à mis à Paul Marillonnet

#4 - 01 avril 2019 11:48 - Benjamin Dauvergne

Utiliser le décorateur

```
@six.python_2_unicode_compatible
```

et renommer tous les `__unicode__` en `__str__`.

#5 - 01 avril 2019 11:48 - Benjamin Dauvergne

- Statut changé de Solution proposée à En cours

#6 - 24 avril 2019 14:54 - Paul Marillonnet

- Fichier `0001-python3-py2-unicode-compatible-__str__-magid-methods.patch` ajouté
- Statut changé de En cours à Solution proposée

#7 - 24 avril 2019 18:18 - Benjamin Dauvergne

Ne va pas marcher en python2:

```
list_display = ('__str__', 'slug', 'ou', 'service', 'admin_scope')
```

C'est bien unicode qu'il faut.

#8 - 24 avril 2019 18:19 - Benjamin Dauvergne

- Statut changé de Solution proposée à En cours

#9 - 25 avril 2019 15:01 - Paul Marillonnet

- Statut changé de En cours à Information nécessaire

Je dois mal comprendre la doc¹ :

On Python 2, it aliases the `__str__` method to `__unicode__` and creates a new `__str__` method that returns the result of `__unicode__()` encoded with UTF-8

Pour moi c'est pas grave d'afficher dans l'interface admin la valeur encodée en python2, plutôt que de se manger un `AttributeError` en python3 parce que `__unicode__` n'existe pas. Je me plante ?

¹https://pythonhosted.org/six/#six.python_2_unicode_compatible

#10 - 25 avril 2019 16:14 - Benjamin Dauvergne

- Statut changé de Information nécessaire à Solution validée

Si tu mets `str` ça va appeler `str` qui renvoie un str encodé en utf-8, Django ne va pas aimer. Si tu veux supporter py2 et py3 tu vas être obligé de mettre un:

```
if six.PY3:
    list_display.remove('__unicode__')
    list_display.insert(0, '__str__')
```

à moins qu'il existe un moyen un peu uniforme de désigner la conversion en text au niveau de `list_display`¹ qui fonctionn en python2 et python3.

¹https://docs.djangoproject.com/fr/1.11/ref/contrib/admin/#django.contrib.admin.ModelAdmin.list_display

PS: je viens de regarder et en fait Django prend en charge la conversion², tu peux utiliser `__str__`

²<https://github.com/django/django/blob/1.11.20/django/contrib/admin/utils.py#L292>

#11 - 02 mai 2019 10:53 - Paul Marillonnet

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

```
commit 43769fb2a7d47ae866146a1d12e65f7df0b7cf02
Author: Paul Marillonnet <pmarillonnet@entrouvert.com>
Date: Thu Feb 21 17:32:53 2019 +0100
```

```
python3: py2 unicode compatible __str__ magid methods (#31184)
```

#12 - 05 mai 2019 17:15 - Frédéric Péters

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

Fichiers

0001-python3-ou.__str__-magid-method-uniformization-31184.patch	791 octets	06 mars 2019	Paul Marillonnet
0001-python3-py2-unicode-compatible-__str__-magid-methods.patch	18,8 ko	24 avril 2019	Paul Marillonnet