

# EOPayment - Development #43082

## traductions

19 mai 2020 15:23 - Frédéric Péters

<b>Statut:</b>	Fermé	<b>Début:</b>	19 mai 2020
<b>Priorité:</b>	Bas	<b>Echéance:</b>	
<b>Assigné à:</b>	Frédéric Péters	<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>			
Il y a des appels à gettext mais pas de .po créés, pas de traductions réalisées.			
Voir également comment ça va pouvoir être derrière exploité par les utilisateurs, que le domaine .po soit utilisé; peut-être qu'il faut modifier get_parameters() pour prendre un paramètre lang et retourner des chaînes traduites. (?)			
<b>Demandes liées:</b>			
Lié à Combo - Bug #45183: Traductions manquantes sur l'ajout d'une plateforme...		<b>Fermé</b>	<b>16 juillet 2020</b>
Lié à Combo - Development #48636: afficher le help_text des paramètres d'un b...		<b>Fermé</b>	<b>18 novembre 2020</b>
Dupliqué par EOPayment - Development #10006: corrections à la gestion des tra...		<b>Fermé</b>	<b>16 février 2016</b>

## Révisions associées

### Révision 6fe18d4d - 19 novembre 2020 09:05 - Frédéric Péters

add support for translation when used as a django app (#43082)

## Historique

### #1 - 16 juillet 2020 19:20 - Frédéric Péters

- Lié à Bug #45183: Traductions manquantes sur l'ajout d'une plateforme de paiement ajouté

### #2 - 11 octobre 2020 09:37 - Benjamin Dauvergne

- Priorité changé de Normal à Bas

On ne sais juste pas faire pour l'instant (intégration localisation Django avec localisation LC\_MESSAGE), si quelqu'un a une solution propre et jolie qu'il la pose ici.

### #3 - 11 octobre 2020 09:45 - Benjamin Dauvergne

- Dupliqué par Development #10006: corrections à la gestion des traductions ajouté

### #4 - 03 novembre 2020 19:01 - Benjamin Dauvergne

Commentaire recopié de [#45183](#).

Je ne sais pas comment ça va marcher; Django ne change pas la locale globale (LANG) donc le un simple gettext.gettext ne marchera pas même si on installe le catalogue (et si on se met à utiliser deux APIs gettext, celle de Django qui utilise ses catalogues et celle de Python). L'autre possibilité c'est de passer explicitement la locale à une méthode get\_description(lang=None) et que ça renvoie des descriptions déjà traduites (et en unicode).

J'ai cherché un peu la marche à suivre concernant des modules python pour les traductions mais ça suppose qu'on utilise les locales normalement pas comme Django.

Un plan pour qui n'en veut :

- produire un fichier de traduction, s'arranger pour l'installer dans /usr/share/locale/<lang>/LC\_MESSAGES/eopayment.mo (ou alors dans le code du projet comme Django, ça ne change pas grand chose), on doit pouvoir réutiliser django-admin makemessages en faisant un répertoire eopayment/locale/fr/LC\_MESSAGES/
- ajouter un paramètre lang=None à Payment.get\_parameters()
- si lang=None, on fait rien
- sinon :
  - on obtient un objet de traduction via gettext.translation('eopayment', [localedir=os.path.join(os.path.dirname(\_\_file\_\_), 'locale')]) (localedir si

- on utilise pas le répertoire système)
- on traverse parameters en traduisant toutes les clés 'caption', 'help\_text' et 'choices' avec trans.gettext(message) (où trans.ugettext(message) pour py2 mais on peut aussi se dire qu'on s'en fout)

PS: penser à ajouter des N\_(..) partout

#### #5 - 18 novembre 2020 10:26 - Frédéric Péters

Proposition alternative pour marcher très facilement/rapidement avec combo, sans poser de problème ailleurs,

Taper dans common.py

```
+try:
+    from django.utils.translation import ugettext_lazy as _
+except ImportError:
+    from gettext import gettext as _ # compat avec un existant bidon, _ = lambda x: x ferait pareil.
```

Remplacer les from gettext import gettext as \_ par des from .common import \_.

Ajouter un répertoire eopayment/locale/fr/LC\_MESSAGES/. Ajouter un appel à django-admin compilemessages -l fr dans le setup.py.

Dans combo ajouter eopayment à l'INSTALLED\_APPS.

#### #6 - 18 novembre 2020 10:26 - Frédéric Péters

- Lié à Development #48636: afficher le help\_text des paramètres d'un backend de paiement ajouté

#### #7 - 18 novembre 2020 18:51 - Benjamin Dauvergne

Frédéric Péters a écrit :

Taper dans common.py

[...]

Ok mais ajouter en plus comme condition 'django' in sys.modules. Sans cela on aura une trace dès que django est installé sur la machine mais n'est pas utilisé au moment où on charge eopayment (genre un test de paiement payfip\_ws avec l'outil CLI intégré) :

```
Python 3.8.6 (default, Sep 25 2020, 09:36:53)
[GCC 10.2.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import sys.modules
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ModuleNotFoundError: No module named 'sys.modules'; 'sys' is not a package
>>> import django.utils.translation
>>> from django.utils.translation import ugettext_lazy as _
>>> _('xxx')
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
  File "/usr/lib/python3/dist-packages/django/utils/functional.py", line 123, in __repr__
    return repr(self.__cast())
  File "/usr/lib/python3/dist-packages/django/utils/functional.py", line 168, in __cast
    return self.__text_cast()
  File "/usr/lib/python3/dist-packages/django/utils/functional.py", line 156, in __text_cast
    return func(*self.__args, **self.__kw)
  File "/usr/lib/python3/dist-packages/django/utils/translation/__init__.py", line 79, in gettext
    return _trans.gettext(message)
  File "/usr/lib/python3/dist-packages/django/utils/translation/__init__.py", line 54, in __getattr__
    if settings.USE_I18N:
  File "/usr/lib/python3/dist-packages/django/conf/__init__.py", line 79, in __getattr__
    self._setup(name)
  File "/usr/lib/python3/dist-packages/django/conf/__init__.py", line 60, in _setup
    raise ImproperlyConfigured(
django.core.exceptions.ImproperlyConfigured: Requested setting USE_I18N, but settings are not configured. You must either define the environment variable DJANGO_SETTINGS_MODULE or call settings.configure() before accessing settings.
```

#### #8 - 18 novembre 2020 20:27 - Frédéric Péters

- Statut changé de Nouveau à En cours

- Assigné à mis à Frédéric Péters

(je viens de pousser une branche de test)

**#9 - 18 novembre 2020 21:12 - Frédéric Péters**

- Fichier 0001-add-support-for-translation-when-used-as-a-django-ap.patch ajouté
- Statut changé de *En cours* à *Solution proposée*
- Patch proposed changé de *Non* à *Oui*

Dans la branche un second patch, avec le .po de la traduction.

Dans le patch attaché ce que je décris ci-dessus, avec la remarque de Benjamin concernant 'django' in sys.modules, ainsi qu'un tour des différents backends de paiement pour retirer le marqueur de traduction des chaînes écrites en français.

**#10 - 18 novembre 2020 23:26 - Benjamin Dauvergne**

- Statut changé de *Solution proposée* à *Solution validée*

**#11 - 19 novembre 2020 09:07 - Frédéric Péters**

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

```
commit 6fe18d4d58a67327cebb1b4707225dc1b086c3c7
Author: Frédéric Péters <fpeters@entrouvert.com>
Date:   Wed Nov 18 20:16:46 2020 +0100
```

```
    add support for translation when used as a django app (#43082)
```

(en local c'était correct mais contenu du .deb produit à vérifier quand même).

**#12 - 19 novembre 2020 21:17 - Frédéric Péters**

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

**Fichiers**

---

0001-add-support-for-translation-when-used-as-a-django-ap.patch 19,6 ko 18 novembre 2020

Frédéric Péters