

Passerelle - Development #46611

Permettre de charger les fichiers urls depuis une application commune, tout en gardant les fonctionnalités liées aux connecteurs qui dérivent de cette application.

14 septembre 2020 13:23 - Nicolas Roche

Statut:	Fermé	Début:	14 septembre 2020
Priorité:	Normal	Echéance:	
Assigné à:	Nicolas Roche	% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	Non
Patch proposed:	Oui		
Description			
J'aurais besoin de ce patch pour charger sms/urls.py sur les connecteurs SMS, pour implémenter un bouton de test d'envoi des SMS (#45333).			
Demandes liées:			
Lié à Passerelle - Development #45333: intégrer au niveau des connecteurs SMS...		Fermé	21 juillet 2020

Révisions associées

Révision a5ee5cb6 - 15 octobre 2020 16:07 - Nicolas Roche

base: load urls.py from generic connectors apps (#46611)

Historique

#1 - 14 septembre 2020 13:33 - Nicolas Roche

- Fichier 0001-base-load-urls.py-from-sms-directory-46611.patch ajouté
- Patch proposed changé de Non à Oui

Je pose un patch, mais c'est uniquement pour clarifier ma description du besoin qui n'est certainement pas claire du tout.

#2 - 14 septembre 2020 14:35 - Frédéric Péters

Tu ne devrais pas aller taper du code spécifique SMS dans passerelle/base/.

L'approche devrait être d'avoir dans passerelle/sms/ le SmsAppConfigMixin et dans la classe modèle de base un attribut, genre app_config_extra_mixins = (SmsAppConfigMixin,).

De là, tu peux avoir dans passerelle/base/ du code qui ferait un générique

```
app.__class__.__bases__ = connector.app_config_extra_mixins + app.__class__.__bases__
```

#3 - 16 septembre 2020 18:06 - Nicolas Roche

- Fichier 0001-base-load-urls.py-from-generic-connectors-apps-46611.patch ajouté
- Statut changé de Nouveau à Solution proposée

J'ai l'impression que si j'ajoute le mixin SMS dans app.__class__.__bases__, alors toutes les applications qui utilisent ConnectorAppConfig vont également bénéficier du mixin. Et donc qu'il me faut plutôt remplacer app.__class__ par une classe qui la dérive.

J'ai déplacé le code dans apps/sms/__init__.py, et ConnectorAppMixin dans base/mixins.py pour éviter l'import circulaire.

En vrai, il n'y a rien ici pour tester ce patch. Pour cela il faut voir la branche posée sur [#45333](#) qui l'utilise.

#4 - 16 septembre 2020 21:34 - Frédéric Péters

- Statut changé de Solution proposée à En cours
- Patch proposed changé de Oui à Non

Ce n'est pas ça que je suggérais mais oublions tout à fait là je vais encore suggérer autre chose : arrêter avec les AppConfig qui introduisent une

hiérarchie parallèle et conduisent à ce ticket.

Rester au truc simple de un connecteur = la classe de son modèle.

Et du coup modifier register_apps_urls pour également itérer sur les modèles.

#5 - 17 septembre 2020 14:50 - Nicolas Roche

- Fichier 0001-base-load-urls.py-from-generic-connectors-apps-46611.patch ajouté

- Statut changé de En cours à Solution proposée

- Patch proposed changé de Non à Oui

Et du coup modifier register_apps_urls pour également itérer sur les modèles.

Aïe, mon code est super moche et du coup j'intuite que je suis à nouveau passé à côté :
j'itère sur la classe qui se serait éventuellement intercalée entre le connecteur et BaseResource.

#6 - 17 septembre 2020 15:24 - Frédéric Péters

Sur l'appconfig tu as un get_connector_model. Il faut utiliser ça.

C'est verbeux à cause du changement d'indentation, mais en gros j'imaginai arriver à du code comme ça :

```
diff --git a/passerelle/plugins.py b/passerelle/plugins.py
index 491805c4..f26d93e9 100644
--- a/passerelle/plugins.py
+++ b/passerelle/plugins.py
@@ -27,29 +27,32 @@ def register_apps_urls(urlpatterns):
     before_urls = []
     after_urls = []
     for app in apps.get_app_configs():
-         if hasattr(app, 'get_before_urls') or hasattr(app, 'get_after_urls'):
-             if hasattr(app, 'get_before_urls'):
-                 urls = app.get_before_urls()
+         if not hasattr(app, 'get_connector_model'):
+             continue
+         connector_model = app.get_connector_model()
+         connector_slug = connector_model.get_connector_slug()
+         for obj in (connector_model, app):
+             if hasattr(obj, 'get_before_urls') or hasattr(obj, 'get_after_urls'):
+                 if hasattr(obj, 'get_before_urls'):
+                     urls = obj.get_before_urls()
+                     if urls:
+                         before_urls.append(url('^', include(urls)))
+                 if hasattr(obj, 'get_after_urls'):
+                     urls = obj.get_after_urls()
+                     if urls:
+                         after_urls.append(url('^', include(urls)))
+             elif hasattr(obj, 'get_urls'):
+                 url_prefix = '^%s/' % connector_slug
+                 urls = obj.get_urls()
+                 if urls:
+                     before_urls.append(url('^', include(urls)))
-         if hasattr(app, 'get_after_urls'):
-             urls = app.get_after_urls()
+                 urls = required(app_enabled(app.label), urls)
+                 after_urls.append(url(url_prefix, include(urls), kwargs={'connector': connector_slug}))
+         if hasattr(obj, 'get_management_urls'):
+                 url_prefix = '^manage/%s/' % connector_slug
+                 urls = obj.get_management_urls()
+                 if urls:
+                     after_urls.append(url('^', include(urls)))
-         elif hasattr(app, 'get_urls'):
-             connector_slug = app.get_connector_model().get_connector_slug()
-             url_prefix = '^%s/' % connector_slug
-             urls = app.get_urls()
-             if urls:
-                 urls = required(app_enabled(app.label), urls)
-                 after_urls.append(url(url_prefix, include(urls), kwargs={'connector': connector_slug}))
-         if hasattr(app, 'get_management_urls'):
-             connector_slug = app.get_connector_model().get_connector_slug()
```

```

-         url_prefix = '^manage/%s/' % connector_slug
-         urls = app.get_management_urls()
-         if urls:
-             urls = required(app_enabled(app.label), urls)
-             urls = required(manager_required, urls)
-             after_urls.append(url(url_prefix, include(urls), kwargs={'connector': connector_slug}))
+         urls = required(app_enabled(app.label), urls)
+         urls = required(manager_required, urls)
+         after_urls.append(url(url_prefix, include(urls), kwargs={'connector': connector_slug}))

    return before_urls + urlpatterns + after_urls

```

(pas testé du tout)

#7 - 17 septembre 2020 16:33 - Nicolas Roche

- Fichier 0001-base-load-urls.py-from-generic-connectors-apps-46611.patch ajouté

Waouh, ça marche du premier coup ! (je pose le patch mais c'est ligne pour ligne le diff ci-dessus)
Et donc ensuite (cf la branche posée sur [#45333](#)), juste ajouter cette méthode de classe dans sms/models.py ?

```

class SMSResource(BaseResource):
    ...
    @classmethod
    def get_management_urls(cls):
        return import_string('passerelle.sms.urls.management_urlpatterns')

```

#8 - 18 septembre 2020 16:27 - Nicolas Roche

- Lié à Development [#45333](#): intégrer au niveau des connecteurs SMS la possibilité d'un envoi de test ajouté

#9 - 18 septembre 2020 17:38 - Nicolas Roche

- Assigné à mis à Nicolas Roche

#10 - 14 octobre 2020 16:12 - Valentin Deniaud

- Statut changé de Solution proposée à Solution validée

Il y a pas de branche mais je vois que c'est vert via [#45333](#), donc go.

#11 - 15 octobre 2020 16:08 - Nicolas Roche

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

```

commit a5ee5cb6e06485ec498957d40c9ff54449eba9b4
Author: Nicolas ROCHE <nroche@entrouvert.com>
Date: Thu Sep 17 16:22:30 2020 +0200

```

```

base: load urls.py from generic connectors apps (#46611)

```

#12 - 16 octobre 2020 01:16 - Frédéric Péters

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

Fichiers

0001-base-load-urls.py-from-sms-directory-46611.patch	2,7 ko 14 septembre 2020	Nicolas Roche
0001-base-load-urls.py-from-generic-connectors-apps-46611.patch	4,1 ko 16 septembre 2020	Nicolas Roche
0001-base-load-urls.py-from-generic-connectors-apps-46611.patch	2,48 ko 17 septembre 2020	Nicolas Roche
0001-base-load-urls.py-from-generic-connectors-apps-46611.patch	3,61 ko 17 septembre 2020	Nicolas Roche