

Passerelle - Development #47161

Développer le connecteur passerelle <> Solis AFI MSS (aides financières Ministère solidarité santé)

30 septembre 2020 10:09 - Nicolas Roche

Statut:	Fermé	Début:	30 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			
L'API est décrite ici : https://dev.entrouvert.org/projects/ministere-des-solidarites-et-de-la-sante/wiki/Solis_de_test			

Révisions associées

Révision ad276bfd - 19 octobre 2020 12:19 - Nicolas Roche

solis_afi_mss: add solis AFI MSS connector (#47161)

Historique

#2 - 01 octobre 2020 12:11 - Nicolas Roche

- Fichier 0001-solis_afi_mss-add-solis-AFI-MSS-connector-47161.patch ajouté
- Tracker changé de Support à Bug
- Statut changé de Nouveau à Solution proposée
- Patch proposed changé de Non à Oui

#3 - 01 octobre 2020 12:31 - Thomas Noël

- Tracker changé de Bug à Development

#4 - 01 octobre 2020 12:58 - Thomas Noël

from django.utils.six... : on est en Python3 maintenant, je crois que les "six" ne doivent plus apparaître dans nos nouveaux codes.

« base_url = models.CharField(max_length=128, blank=False, verbose_name=_('url')) » : il faut être plus explicite que juste "url" et préciser ce qui est attendu ici. (éventuellement en posant un exemple). Même chose sur description=_('Demands'), ou description=_('Demand'), etc : il faut ajouter des verbe et expliciter. Par exemple "Créer un nouvelle demande", "obtenir la liste des demandes en cours", etc. (en anglais, tu sais mieux faire que moi ;))

Sur les schemas TAX_SCHEMA et DEMAND_SCHEMA :

- je préférerais qu'on garde les noms Solis pour les variables. Ca va simplifier la doc.
- attention, wcs n'envoie que des chaines de caractères, dans DEMAND_SCHEMA il faut spécifier autre chose pour "index_list". Je propose individusConcernes contenant une liste d'id séparée par des ":",

Perso je pense qu'on ne devrait pas faire trop de validation des inputs mais envoyer quasi directement les données JSON reçues de wcs vers Solis. Comme ça si le schéma Solis évolue il n'y aura pas besoin d'adapter notre connecteur. Ainsi pour DEMAND_SCHEMA :

- tu mets tous les champs optionnels, au cas où ils disparaissent un jour de Solis...
- tu permets de recevoir des champs supplémentaires (je crois que c'est déjà le cas par défaut, ceci dit)
- tu spécifierais juste le format qu'on attendrait pour individusConcernes (genre une liste d'id séparée par des ":",)
- dans le code, tu modifies juste post_data["individusConcernes"] pour y poser les individus concernés, et s'il n'y a pas d'index

Pareil ailleurs, par exemple quand tu fais un "if code not in ['2', '6']:" on s'en fiche, on envoie la demande à Solis, c'est lui qui vérifiera que c'est 2 ou 6. Passerelle reste d'abord une passerelle, on doit y minimiser l'intelligence.

Dans get_family je vois une gestion du cache que je ne comprends pas, puisque tu fais systématiquement la recherche avant de cacher la valeur :

```
# cache index for an hour
cache_key = 'solis-afi-mss-%s-index-%s' % (self.id, email)
index = response.json().get('indexAgent')
assert index
cache.set(cache_key, index, 3600)
```

Voilà pour une première lecture rapide, je te laisse déjà répondre ce que tu penses de ma demande sur les schémas.

#5 - 01 octobre 2020 16:44 - Nicolas Roche

- Fichier 0001-solis_afi_mss-add-solis-AFI-MSS-connector-47161.patch ajouté

Remarques presque toutes prises en compte :

Tu spécifierais juste le format qu'on attendrait pour individusConcernes

Ok, j'ai tout mis dans des chaînes et commenté toutes les autres regex.

Tu permets de recevoir des champs supplémentaires (je crois que c'est déjà le cas par défaut, ceci dit)

Oui, ça passe avec des champs supplémentaires sans avoir rien changé.

Dans le code, tu modifies juste `post_data['individusConcernes']` pour y poser les individus concernés, et s'il n'y a pas d'index

(je n'ai pas compris ici, à priori passer les index doit suffire puisque les autres champs proviennent déjà de chez eux)

Je vois une gestion du cache que je ne comprends pas

J'ai simplifié

#6 - 01 octobre 2020 17:30 - Nicolas Roche

- Fichier 0002-solis_afi_mss-translation-update-47161.patch ajouté

- Fichier 0001-solis_afi_mss-add-solis-AFI-MSS-connector-47161.patch ajouté

(j'ai traduit puisque les chaînes en français étaient données dans la doc)

#7 - 02 octobre 2020 01:48 - Thomas Noël

Ok, j'ai tout mis dans des chaînes et commenté toutes les autres regex.

Je n'ai pas compris pourquoi tu as commenté les pattern. Autant les supprimer ? Je pense aussi qu'il n'est pas utile de tout mettre en "string", tu peux laisser integer et number quand on sait que c'est ce qui est attendu, car je crois que l'analyseur sait comprendre des nombres reçus sous forme de string (à tester).

Quelques autres points :

individusConcernes : il faut que le pattern soit un peu plus souple, on aura éventuellement du mal à avoir un "join" en Django, laisse la possibilité d'avoir des ":" en trop : « `r'^[0-9]*(:[0-9]+:)*$'` » et parse avec un « `[i.strip() for i in individus.split(':') if i.strip()]` »

get_family : il y a une erreur sur le return, on y met juste data, le `err:0` est ajouté automatiquement par le décorateur. En fait je pense qu'il faut que tu sortes le process du "get_family" du endpoint, c-a-d avoir une fonction classique "def get_family_from_email(self, email) et un endpoint "def get_family(...)" qui l'appellera. Et c'est `self.get_family_from_email(email)` que tu appelleras au lieu de `self.get_family..`

text_template='{{prenom}} {{nom}}' je n'ai pas compris l'utilité de laisser cela paramétrable. Pour moi ça sera toujours « `'%(prenom)s %(nom)s' % data` », il y a un besoin autre tu penses ?

post_demand : on ne s'est pas compris, pour moi le payload c'est `post_data`. S'il y a des trucs en plus, on les envoie à Solis. Tu as juste à ajouter un `indexAgent` dans le `post_data` et à transformer `individusConcernes` si c'est une string. Même chose pour `post_tax`

`urljoin(self.base_url + '/', uri)` : l'ajout de / est inutile, c'est justement le travail d'`urljoin` de gérer ça.

`base_url = models.CharField` : plutôt `URLField`. Attention à être généreux sur la longueur, une URL ça s'allonge vite, mets 256 :)

Enfin, plus généralement, j'aime pas tellement les URL avec des "_" donc je pense que tu devrais utiliser la possibilité de nommer les endpoints, afin d'avoir par exemple "get-family" au lieu de "get_family". Je ne suis pas fan non plus des "get" et "post" dans le nom du endpoint, la méthode HTTP précise déjà les choses. D'ailleurs ça manque les `methods=['get']` et `methods=['post']`

(j'arrête ici, dodo :)

#8 - 02 octobre 2020 10:58 - Nicolas Roche

Merci pour la relecture, je prend tout ça en compte asap... Mais d'ici là, est-ce que tu peux préciser ces points stp :

tu peux laisser integer et number quand on sait que c'est ce qui est attendu, car je crois que l'analyseur sait comprendre des nombres reçus sous forme de string (à tester).

Mes tests semblent dire que l'analyseur ne fait pas ça :

```
TAX_SCHEMA = {...
    'montantImposition': {
        'description': 'Tax amount',
        'type': 'number',
    }
}

$ curl 'https://passerelle.dev.publik.love/solis-afi-mss/test/post_tax' --data '{"email": "jacques.rousseau.cts@tiralarc-grand-est.fr", "indexImposition": "222", "anneeImposition": "2020", "nombrePartImposition": "3.2", "montantImposition": "777.77"}'
{"err": 1, "err_class": "passerelle.utils.jsonresponse.APIError", "err_desc": "montantImposition: '777.77' is not of type 'number'", "data": null}
```

`urljoin(self.base_url + '/', uri)` : l'ajout de / est inutile, c'est justement le travail d'urljoin de gérer ça.

Pour moi ça ne prend pas en compte le cas d'une URL de base qui possède déjà un chemin mais qui ne précise pas le slash final :

```
>>> from urllib.parse import urljoin
>>> urljoin('http://localhost/rep1', 'rep2')
'http://localhost/rep2'
>>> urljoin('http://localhost/rep1/', 'rep2')
'http://localhost/rep1/rep2'
```

Je ne suis pas fan non plus des "get" et "post" dans le nom du endpoint, la méthode HTTP précise déjà les choses.

Oui, est-il possible d'utiliser le même nom pour 2 endpoints différents (un en GET et l'autre en POST) ?

#9 - 02 octobre 2020 11:34 - Thomas Noël

Nicolas Roche a écrit :

Merci pour la relecture, je prend tout ça en compte asap... Mais d'ici là, est-ce que tu peux préciser ces points stp :

tu peux laisser integer et number quand on sait que c'est ce qui est attendu, car je crois que l'analyseur sait comprendre des nombres reçus sous forme de string (à tester).

Mes tests semblent dire que l'analyseur ne fait pas ça

Cool merci pour le test, j'aurais dû le faire avant de parler. J'ai dû confondre avec un autre analyseur. Laissons donc en string. Il faudra voir si l'envoi de string ne gêne pas Solis, sinon il faudra caster.

`urljoin(self.base_url + '/', uri)` : l'ajout de / est inutile, c'est justement le travail d'urljoin de gérer ça.

Pour moi ça ne prend pas en compte le cas d'une URL de base qui possède déjà un chemin mais qui ne précise pas le slash final :

Ca m'apprendra à bosser la nuit :) Bien vu again.

Je ne suis pas fan non plus des "get" et "post" dans le nom du endpoint, la méthode HTTP précise déjà les choses.

Oui, est-il possible d'utiliser le même nom pour 2 endpoints différents (un en GET et l'autre en POST) ?

Tu peux regarder mais je pense que non.

#10 - 02 octobre 2020 18:17 - Nicolas Roche

- Fichier 0001-solis_afi_mss-add-solis-AFI-MSS-connector-47161.patch ajouté

- Fichier 0002-solis_afi_mss-translation-update-47161.patch ajouté

(remarques prises en compte)

#11 - 02 octobre 2020 19:13 - Thomas Noël

- `Template('{{prenom}} {{nom}}').render(Context(record)).strip()` ça consomme beaucoup de ressources pour rien, utilise « '%(prenom)s %(nom)s' % record ». D'une façon générale, supprimer tous les `Template()` on n'en a pas besoin ici (on développe pour un seul client, on n'aura pas de travail dérivé à faire)
- ce genre ce code :

```
response = self.requests.get(url, params={'adresseMail': email})
if response.status_code != 200:
    raise APIError('email not found')
```

Ça ne va pas. Si on reçoit une 500 l'erreur est ailleurs, mais l'utilisateur de Passerelle ne va pas le comprendre. Ajoute-toi une méthode `self.request` telle que celle `passerelle/apps/solis/models.py`, tu pourras l'utiliser partout factoriser tous les `self.get_resource_url + self.requests.get` et virer ton "check_reponse"

Bon, en fait je teste en vrai et je vois que leur API crache une 500 en cas d'email non trouvé. Pffff...! Il faut le leur dire, c'est un bogue chez Solis (ticket à faire). On doit recevoir une 200 avec un message d'erreur clair (en json). Ou, au pire, une 404. Mais pas une 500 qui va nous alerter de partout.

- dans `search_by_email` : family tu en fait un liste qui mélange adultes et enfants. A mon avis il ne faut pas faire ça, il faut avoir deux listes, les adultes et les enfants. J'imaginerais plutôt que la fonction finisse avec un "return index, adults, children". Mais bon, je ne sais pas à quoi sert cette liste... donc peut-être que les listes d'adulte et d'enfant ça ne sert à rien non plus ?
- `get_index` ne sert plus à rien, puisqu'un appel à `search_by_email` fait le même travail. Aucun besoin de cache nulle part dans ce connecteur, imho.
- plutôt que des "del post_data['email']" utilise pop :

```
email = post_data.pop('email')
index, adults, children = search_by_email(email)
```

- `get_tax` -> à supprimer et à remplacer par l'ajout d'une option "id" sur `get_taxes` qui remonter la taxe pour l'année "id"
- renommages :
 - `get_taxes` -> taxes
 - `post_tax` -> declare-tax
 - `family_ratio` -> simulate-quotient
 - `demands` -> helps
 - `demand` -> demand-help
- dans taxes et helps, tu peux simplifier :

```
for record in response.json():
    data = {}
    for key, value in record.items():
        data[key] = value
    data['id'] = record.get('anneeImposition')
    template = Template(text_template)
    data['text'] = template.render(Context(data)).strip()
    result.append(data)
```

comme ça :

```
for record in response.json():
    record['id'] = record.get('anneeImposition')
    record['text'] = _('%(anneeImposition)s: %s(montantImposition)s') % record
    result.append(record)
```

#12 - 05 octobre 2020 15:07 - Nicolas Roche

- Fichier `0002-solis_afi_mss-translation-update-47161.patch` ajouté
- Fichier `0001-solis_afi_mss-add-solis-AFI-MSS-connector-47161.patch` ajouté

Remarques prises en compte :

- J'ai viré tous les templates.
- J'ai fait un ticket pour obtenir autre chose que des 500 en cas d'erreurs applicatives : #47338
- J'ai divisé la famille entre adults et enfants, mais je la reconstruit ensuite car le but c'est de présenter une liste multiple dans WCS pour l'utiliser comme paramètre individusConcernes du WS afi/aide/deposer/
- J'ai viré le cache sur l'index de l'agent
- J'utilise "pop" pour retirer les "del" (qui deviennent inutiles)
- get_tax supprimé au profit de get_taxes(year) : qui renvoie alors une listee de 1 ou 0 éléments
- J'ai tout renommé avec des tirets (désolés pour les précédents oublis à propos du renommage)
- J'utilise directement le dictionnaire response.json() pour ajouter et retourner les données

#13 - 05 octobre 2020 15:41 - Nicolas Roche

- Fichier 0002-solis_afi_mss-translation-update-47161.patch ajouté
- Fichier 0001-solis_afi_mss-add-solis-AFI-MSS-connector-47161.patch ajouté

#14 - 05 octobre 2020 15:44 - Nicolas Roche

- Fichier 0001-solis_afi_mss-add-solis-AFI-MSS-connector-47161.patch supprimé

#15 - 05 octobre 2020 15:44 - Nicolas Roche

- Fichier 0002-solis_afi_mss-translation-update-47161.patch supprimé

#16 - 05 octobre 2020 15:44 - Nicolas Roche

- Fichier 0002-solis_afi_mss-translation-update-47161.patch ajouté
- Fichier 0001-solis_afi_mss-add-solis-AFI-MSS-connector-47161.patch ajouté

Ajout de catégories.

#17 - 08 octobre 2020 18:41 - Nicolas Roche

Mais bon, je ne sais pas à quoi sert cette liste...

À construire le paramètre "individusConcernes", à partir d'une liste à choix multiples dans w.c.s. branché sur cette "source de donnée". Ce paramètre est requis pour pousser une demande.

Par exemple, j'ai configuré un appel webservice dans w.c.s. pour tester l'envoi des demandes, en utilisant ces gabarits :

- email: {{form_user_var_email}}
- codeTypeAide: 24
- natureTypeAide: A
- individusConcernes: {% for id in form_var_individus_concernes_raw %}{{id}}: {% endfor %}
- dateDebut: {{form_var_debut|date:"Y-m-d"}}
- dateFin: {{form_var_fin|date:"Y-m-d"}}
- montantFacture: {% localize off %}{{form_var_montant|decimal}}{% endlocalize %}

(je n'ai pas trouvé plus simple pour passer la liste)

#18 - 08 octobre 2020 23:47 - Thomas Noël

Nicolas Roche a écrit :

Mais bon, je ne sais pas à quoi sert cette liste...

À construire le paramètre "individusConcernes", à partir d'une liste à choix multiples dans w.c.s. branché sur cette "source de donnée". Ce paramètre est requis pour pousser une demande.

Mais donc cette liste va mixer adultes et enfants. Ca peut être un poil "bizarre"... Bon, si c'est ça qui est prévu dans le formulaire et que le ministère est ok, tant mieux...

- individusConcernes: {% for id in form_var_individus_concernes_raw %}{{id}}: {% endfor %}
- (je n'ai pas trouvé plus simple pour passer la liste)

```
{{ form_var_individus_concernes_raw|join:";" }}
```

#19 - 09 octobre 2020 09:40 - Nicolas Roche

- Fichier 0001-solis_afi_mss-add-solis-AFI-MSS-connector-47161.patch ajouté

Merci, pour le gabarit.

Oui, le mieux est de laisser le choix de présenter une ou deux liste dans w.c.s.

#20 - 09 octobre 2020 09:55 - Nicolas Roche

- Fichier 0001-solis_afi_mss-add-solis-AFI-MSS-connector-47161.patch ajouté

(en ajoutant 2 nouveaux endpoints "adults" et "children" pour pouvoir les utiliser comme sources de données)

#21 - 09 octobre 2020 09:55 - Nicolas Roche

- Fichier 0001-solis_afi_mss-add-solis-AFI-MSS-connector-47161.patch supprimé

#22 - 09 octobre 2020 13:13 - Thomas Noël

Le code de requête est factorisable, on voit bien la répétition de self.get_resource_url / self.requests / self.check_response

Tu peux plus proprement avoir une fonction "request" qui va faire tout cela d'un coup (plus besoin de get_resource_url et check_response) et renvoyer le JSON reçu si tout va bien. Ca va rendre les endpoints plus faciles à lire. Et check_status se limitera à un simple « self.request('main/isAlive/') »

(et promis je pense qu'avec ça, ça sera un ack ;))

#23 - 09 octobre 2020 18:03 - Nicolas Roche

- Fichier 0002-solis_afi_mss-translation-update-47161.patch ajouté

- Fichier 0001-solis_afi_mss-add-solis-AFI-MSS-connector-47161.patch ajouté

Tu peux plus proprement avoir une fonction "request" qui va faire tout cela d'un coup

oui, j'ai repris celle du connecteur apps/solis (avec #47338 on reçoit maintenant des erreurs formatées)

#24 - 12 octobre 2020 16:39 - Thomas Noël

Ça sent le ack ;)

- 'List of allowance related family member indexes' : précise ici que le séparateur est ".". Je trouve la regex super compliquée : `^[0-9 :]+$` et basta.
- montantFacture tu étais fatigué pour la description ? ;)
- pour base_url tu mets comme exemple <https://solis.mon-application.fr/api-mss-afi/> mais dans le code tu fais `self.base_url + '/'` : il faut choisir ;)
Moi j'enlèverais le `"/` qui est très moche, celui qui configure doit mettre un `/` final dans `base_url`, l'exemple le montrera bien.
- # TO REMOVE AS SOON AS POSSIBLE (#47525) pas utile de le dire ici, puisqu'il y a un ticket ;) ... mais en fait on pourrait considérer que quand `response.text` est vide on renvoie `None` et voilà (c'est pas non plus totalement idiot). Ajuster le test juste au dessus comme ça « `if response.status_code == 204 or not response.text: return None` »

Et deux minis remarques sur le code :

1) Dans :

```
data = []
for record in response['aidesFinancieres']:
    record['id'] = record.get('indexAideFinanciere')
    record['text'] = '%(dateDemandeAide)s (%(suiviAide)s)' % record
    data.append(record)
return {'err': 0, 'data': data}
```

tu n'as pas besoin de data, tu peux directement utiliser `response['aidesFinancieres']` parce qu'en python tout est référence :

```
for record in response['aidesFinancieres']:
    record['id'] = record.get('indexAideFinanciere')
    record['text'] = '%(dateDemandeAide)s (%(suiviAide)s)' % record
return {'err': 0, 'data': response['aidesFinancieres']}
```

2) Ensuite perso j'aurais fait en sorte que `search_from_email` se termine par « `return index, adults, children` » au lieu de s'encombrer d'un dico pas toujours joli à utiliser.

#25 - 13 octobre 2020 11:59 - Nicolas Roche

- Fichier 0001-solis_afi_mss-add-solis-AFI-MSS-connector-47161.patch ajouté

#26 - 19 octobre 2020 11:41 - Thomas Noël

- Statut changé de Solution proposée à Solution validée

#27 - 19 octobre 2020 11:52 - Thomas Noël

S'agissant d'un passerelle.contrib je te propose de pousser ça ce lundi.

Dans la traduction quelques petits trucs :

- "ex: <https://solis.mon-application.fr/api-mss-afi/>" → "exemple : <https://solis.mon-application.fr/api-mss-afi/>"
- "Récupérer la composition familiale d'un agent" → "Récupérer la liste des personnes dans le foyer (adultes et enfants)"
- "Récupérer la composition du couple d'un agent" → "Récupérer la liste des adultes"
- "Récupérer les enfants d'un agent" → "Récupérer la liste des enfants"
- "Récupérer l'adresse fiscale du citoyen" : de l'agent
- "Créer une aide pour un agent" → "Créer une demande d'aide pour un agent"
- "calcul d'un Quotient" → "calcul d'un quotient"
- "nombre de part" → "nombre de parts"

mais tout ça sera dans un patch global de traduction, à faire lors du tag.

#28 - 19 octobre 2020 12:25 - Nicolas Roche

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

```
commit ad276bfdc6aca91760c1d7160a35bbfcf55bb34bb
Author: Nicolas ROCHE <nroche@entrouvert.com>
Date:   Wed Sep 23 18:15:13 2020 +0200
```

```
solis_afi_mss: add solis AFI MSS connector (#47161)
```

#29 - 20 octobre 2020 18:23 - Frédéric Péters

Dans la traduction quelques petits trucs :

mais je ne vois pas de commit de traduction. (?)

mais tout ça sera dans un patch global de traduction, à faire lors du tag.

J'étais parti pour traduire/taguer et puis je me suis trouvé à pas vraiment capter le sens de certains passages; du coup je laisse ça à traduire.

#30 - 21 octobre 2020 01:52 - Thomas Noël

Frédéric Péters a écrit :

J'étais parti pour traduire/taguer et puis je me suis trouvé à pas vraiment capter le sens de certains passages; du coup je laisse ça à traduire.

Yep, je m'en suis occupé. Déployé en recette avec ajout du settings :

```
# /etc/passerelle/settings.d/50solis_afi_mss.py
if 'passerelle.contrib.solis_afi_mss' not in INSTALLED_APPS:
    INSTALLED_APPS += ('passerelle.contrib.solis_afi_mss',)
    TENANT_APPS += ('passerelle.contrib.solis_afi_mss',)
PASSERELLE_APP_SOLIS_AFI_MSS_ENABLED = False
```

et publication uniquement sur le tenant concerné via settings.json avec "PASSERELLE_APP_SOLIS_AFI_MSS_ENABLED": true

#31 - 21 octobre 2020 02:17 - Frédéric Péters

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

Fichiers

0001-solis_afi_mss-add-solis-AFI-MSS-connector-47161.patch	59,4 ko	01 octobre 2020	Nicolas Roche
0001-solis_afi_mss-add-solis-AFI-MSS-connector-47161.patch	60 ko	01 octobre 2020	Nicolas Roche

0002-solis_afi_mss-translation-update-47161.patch	6,44 ko	01 octobre 2020	Nicolas Roche
0001-solis_afi_mss-add-solis-AFI-MSS-connector-47161.patch	60,1 ko	01 octobre 2020	Nicolas Roche
0001-solis_afi_mss-add-solis-AFI-MSS-connector-47161.patch	56,6 ko	02 octobre 2020	Nicolas Roche
0002-solis_afi_mss-translation-update-47161.patch	6,4 ko	02 octobre 2020	Nicolas Roche
0002-solis_afi_mss-translation-update-47161.patch	6,92 ko	05 octobre 2020	Nicolas Roche
0001-solis_afi_mss-add-solis-AFI-MSS-connector-47161.patch	54,6 ko	05 octobre 2020	Nicolas Roche
0002-solis_afi_mss-translation-update-47161.patch	7,25 ko	05 octobre 2020	Nicolas Roche
0001-solis_afi_mss-add-solis-AFI-MSS-connector-47161.patch	55 ko	05 octobre 2020	Nicolas Roche
0001-solis_afi_mss-add-solis-AFI-MSS-connector-47161.patch	57,5 ko	09 octobre 2020	Nicolas Roche
0002-solis_afi_mss-translation-update-47161.patch	19,4 ko	09 octobre 2020	Nicolas Roche
0001-solis_afi_mss-add-solis-AFI-MSS-connector-47161.patch	58,8 ko	09 octobre 2020	Nicolas Roche
0001-solis_afi_mss-add-solis-AFI-MSS-connector-47161.patch	58,5 ko	13 octobre 2020	Nicolas Roche