

Passerelle - Bug #16768

Faire une api commune pour les connecteurs d'état civil

08 juin 2017 16:31 - Josué Kouka

Statut:	Rejeté	Début:	08 juin 2017	
Priorité:	Normal	Echéance:		
Assigné à:	Serghei Mihai	% réalisé:	0%	
Catégorie:		Temps estimé:	0:00 heure	
Version cible:		Planning:		
Patch proposed:	Oui			
Description				
Demandes liées:				
Lié à Passerelle - Development #15883: Connecteur Etat Civil CityWeb		Fermé	14 avril 2017	12 juin 2017
Lié à Passerelle - Bug #16646: mdel: factoriser les outils xml utilisé par le...		Rejeté	31 mai 2017	
Lié à Passerelle - Autre #17477: Spécification API "connecteur état civil"		Rejeté	08 juillet 2017	

Historique

#1 - 08 juin 2017 16:31 - Josué Kouka

- Lié à Development #15883: Connecteur Etat Civil CityWeb ajouté

#2 - 08 juin 2017 16:32 - Josué Kouka

- Lié à Bug #16646: mdel: factoriser les outils xml utilisé par le connecteur. ajouté

#3 - 08 juin 2017 22:27 - Frédéric Péters

Précédemment l'idée était plutôt d'avoir la même API, plutôt qu'un connecteur "générique". (à la manière de l'API SMS, de l'API factures, de l'API famille).

#4 - 10 juin 2017 10:14 - Josué Kouka

- Description mis à jour

- Sujet changé de Faire un connecteur générique état civil à Faire une api commune pour les connecteurs d'état civil

#5 - 10 juin 2017 10:14 - Josué Kouka

- Statut changé de Nouveau à En cours

#6 - 18 juin 2017 18:52 - Serghei Mihai

- Fichier 0001-add-civil_status-generic-API-interface-16768.patch ajouté

- Patch proposed changé de Non à Oui

Ok, donc une classe exposant les endpoints create et status permettant de créer et récupérer le statut d'un dossier respectivement.

La liste des paramètres obligatoires varie d'un connecteur à l'autre donc ici il faut vérifier les paramètres communs. Pour l'instant je n'ai identifié que demand_id.

Contrairement à ce que fait MDEL, je suis d'avis que les paramètres passés au connecteur doivent être dans le payload json, sans extra. Cela aide aussi à la compréhension de ce que est envoyé au connecteur dans l'appel webservice de wcs.

#7 - 18 juin 2017 19:11 - Frédéric Péters

Ce serait pas mal d'explicitement un plan dans ce ticket; on passe de "connecteur générique" à "API commune" via un changement de titre du ticket, sans détails, et là ça partirait quand même toutefois vers une part de code commun, avec déjà pour un endpoint un conflit explicite avec l'API existant dans le connecteur MDEL, ce qui amènera une grande joie lorsqu'à la mise à jour de passerelle il faudra en même temps aller éditer des workflows.

Un plan, ça pourrait commencer par réfléchir et définir précisément une API (genre dans une page du wiki, pas sous forme de code), qui aurait notamment comme caractéristique de ne pas avoir de endpoint en conflit avec ceux du connecteur MDEL, que ce connecteur puisse ensuite être étendu pour gérer à la fois son API et la nouvelle.

#8 - 18 juin 2017 19:28 - Serghei Mihai

Il y a un début de page décrivant l'API générique: https://dev.entrouvert.org/projects/passerelle/wiki/Connecteur_Etat_Civil ou il y a encore pas mal de choses à rajouter.

#9 - 18 juin 2017 19:39 - Josué Kouka

Serghei Mihai a écrit :

Ok, donc une classe exposant les endpoints create et status permettant de créer et récupérer le statut d'un dossier respectivement. La liste des paramètres obligatoires varie d'un connecteur à l'autre donc ici il faut vérifier les paramètres communs. Pour l'instant je n'ai identifié que demand_id.

Contrairement à ce que fait MDEL, je suis d'avis que les paramètres passés au connecteur doivent être dans le payload json, sans extra. Cela aide aussi à la compréhension de ce que est envoyé au connecteur dans l'appel webservice de wcs.

Normalment le demand_id est déduit du display_id envoyé dans le payload.

Il y a un début de page décrivant l'API générique: https://dev.entrouvert.org/projects/passerelle/wiki/Connecteur_Etat_Civil ou il y a encore pas mal de choses à rajouter.

Oui oui.

A priori les paramètres obligatoires seront:

- display_id et receipt_time qui sont envoyé dans le payload
- le type de la demande (Naissance, Mariage, Décès, ...) et le type de document (Copie Intégrale, Extrait avec Filiation, Extrait sans filiation, ...)
- Noms et prénoms du demandeur
- Noms et prénoms de l'intéressé

Je vais finir l'édition de la page.

#10 - 20 juin 2017 11:34 - Serghei Mihai

Je suis d'avis d'avoir les paramètres sur les actes, le demandeur et l'interesse en français, genre:

- type_demande
- type_document
- prenom_demandeur
- nom_demandeur
- prenom_interesse
- nom_interesse

pour pouvoir les publier dans la doc de l'admin fonctionnel et ainsi que les clients/CPF soient autonomes dans la configuration de l'appel au connecteur d'état civil.

Et au passage utiliser plutôt "date_reception" et non "receipt_time". Le display_id peut rester tel quel.

#11 - 07 juillet 2017 14:00 - Josué Kouka

- Fichier 0001-add-civil-status-common-api-16768.patch ajouté

#12 - 07 juillet 2017 14:16 - Serghei Mihai

Dans mon esprit on ne prend pas en compte les extras: tous les paramètres, documentés, doivent être dans un dico plat et lors de l'appel au webservice on leur attribue les bonnes valeurs issues du formulaire.

#13 - 07 juillet 2017 15:51 - Thomas Noël

Serghei Mihai a écrit :

Dans mon esprit

Dans le mien aussi, mode "kiss" avec emphase sur "stupid".

#14 - 07 juillet 2017 16:13 - Josué Kouka

- Fichier 0001-add-civil-status-common-api-16768.patch ajouté

Thomas Noël a écrit :

Serghei Mihai a écrit :

Dans mon esprit

Dans le mien aussi, mode "kiss" avec emphase sur "stupid".

Ok.

#15 - 07 juillet 2017 16:23 - Frédéric Péters

formdata = self.check_keys(formdata) laisse trop penser que check_keys pourrait modifier formdata.

if (key not in formdata) and (key not in formdata.get('fields', {})): la référence à 'fields' me laisse penser qu'on n'en est pas encore à juste considérer les données explicitement transmises, qu'il y a encore un jeu avec la transmission automatique des champs dans w.c.s.

À ce titre-là, aussi, les suffixes `_raw` dans l'API semblent superflus.

Aussi, avant ce code, n'y aurait-il pas besoin de relecture sur l'API proposée ? (pour ma part j'ai juste noté en passant un truc, qui a été pris en compte sans mise à jour de l'exemple, mais je n'ai pas eu de lecture détaillée).

#16 - 07 juillet 2017 18:06 - Josué Kouka

- Fichier `0001-add-civil-status-common-api-16768.patch` ajouté

Frédéric Péters a écrit :

formdata = self.check_keys(formdata) laisse trop penser que check_keys pourrait modifier formdata.

+1

if (key not in formdata) and (key not in formdata.get('fields', {})): la référence à 'fields' me laisse penser qu'on n'en est pas encore à juste considérer les données explicitement transmises, qu'il y a encore un jeu avec la transmission automatique des champs dans w.c.s.

+1

À ce titre-là, aussi, les suffixes `_raw` dans l'API semblent superflus.

+1

Aussi, avant ce code, n'y aurait-il pas besoin de relecture sur l'API proposée ? (pour ma part j'ai juste noté en passant un truc, qui a été pris en compte sans mise à jour de l'exemple, mais je n'ai pas eu de lecture détaillée).

Je pense que si.

J'ai juste mis dans ce patch les clés obligatoires (selon moi)

#17 - 08 juillet 2017 20:46 - Frédéric Péters

À ce titre-là, aussi, les suffixes `_raw` dans l'API semblent superflus.

+1

Mais la page des spécifications contient encore des `_raw`.

Aussi, avant ce code, n'y aurait-il pas besoin de relecture sur l'API proposée ? (pour ma part j'ai juste noté en passant un truc, qui a été pris en compte sans mise à jour de l'exemple, mais je n'ai pas eu de lecture détaillée).

Je pense que si.

J'ai donc créé [#17477](#) pour cette relecture.

#18 - 17 juillet 2017 10:11 - Josué Kouka

- Lié à Autre [#17477](#): Spécification API "connecteur état civil" ajouté

#19 - 27 juillet 2017 14:59 - Josué Kouka

- Fichier `0001-add-civil-status-common-api-16768.patch` ajouté

Une version de l'api commune avec la version actuelle de la doc

#20 - 27 juillet 2017 15:16 - Frédéric Péters

Une version de l'api commune avec la version actuelle de la doc

/<connector>/<slug>/status/<demand_id>/ dans la documentation, je ne vois pas comment ce patch gère ça.

Je trouve toujours très peu d'utilité à cette écriture de code "mis en commun", particulièrement quand il n'est pas utilisé.

Faire :

```
-class MDEL(BaseResource):  
+class MDEL(CivilStatusMixin, BaseResource):
```

Pour prouver que ça a un sens.

Et j'écrivais plus haut :

[...] déjà pour un endpoint un conflit explicite avec l'API existant dans le connecteur MDEL, ce qui amènera une grande joie lorsqu'à la mise à jour de passerelle il faudra en même temps aller éditer des workflows.

Ça sera nettement révélé ici.

#21 - 27 juillet 2017 19:04 - Josué Kouka

Frédéric Péters a écrit :

Une version de l'api commune avec la version actuelle de la doc

/<connector>/<slug>/status/<demand_id>/ dans la documentation, je ne vois pas comment ce patch gère ça.

Je trouve toujours très peu d'utilité à cette écriture de code "mis en commun", particulièrement quand il n'est pas utilisé.

Faire :

[...]

Pour prouver que ça a un sens.

Et j'écrivais plus haut :

[...] déjà pour un endpoint un conflit explicite avec l'API existant dans le connecteur MDEL, ce qui amènera une grande joie lorsqu'à la mise à jour de passerelle il faudra en même temps aller éditer des workflows.

Ça sera nettement révélé ici.

J'avais cru comprendre que le plan était de spécifier/développer l'api commune. L'utiliser avec CityWeb puis par la suite faire un refactoring du connecteur MDEL pour qu'il l'utilise ensuite.

Faire un refactoring de MDEL en meme temps ne me dérange pas non plus. On va partir dessus sauf avis contraire.

#22 - 27 juillet 2017 19:52 - Frédéric Péters

J'avais cru comprendre que le plan [...]

Il y avait une invitation dans un commentaire plus tôt : « Ce serait pas mal d'explicitier un plan [...] ».

Sans nécessairement toucher plus avant au code du connecteur mdel dès maintenant, lui faire utiliser la classe, ça permettra déjà d'éviter les clashes de compatibilité, qui rendront la migration pénible. (ce que je notais "déjà pour un endpoint un conflit explicite avec l'API existant dans le connecteur MDEL, ce qui amènera une grande joie lorsqu'à la mise à jour de passerelle il faudra en même temps aller éditer des workflows.")

#23 - 01 août 2017 10:43 - Josué Kouka

- Fichier 0001-add-civil-status-common-api-16768.patch ajouté

Frédéric Péters a écrit :

J'avais cru comprendre que le plan [...]

Il y avait une invitation dans un commentaire plus tôt : « Ce serait pas mal d'expliciter un plan [...] ».

Sans nécessairement toucher plus avant au code du connecteur mdel dès maintenant, lui faire utiliser la classe, ça permettra déjà d'éviter les clashes de compatibilité, qui rendront la migration pénible. (ce que je notais "déjà pour un endpoint un conflit explicite avec l'API existant dans le connecteur MDEL, ce qui amènera une grande joie lorsqu'à la mise à jour de passerelle il faudra en même temps aller éditer des workflows.")

Avec l'utilisation de la mixin par le connecteur MDEL. Pas de conflit pour l'instant parce que les endpoints create et status sont surchargés par le code actuel du connecteur MDEL.

#24 - 07 août 2017 18:54 - Josué Kouka

- Fichier 0002-make-MDEL-CivilStatusMixin-compliant-16768.patch ajouté

- Fichier 0001-add-civil-status-common-api-16768.patch ajouté

Par rapport aux autres patches, dans celui ci CivilStatusMixin est utilisé par MDEL.

J'ai rajouté un test pour m'assurer que les payload au nouveau et au format "legacy" sont correctement traités.

Pareil pour l'interrogation du endpoint status.

#25 - 08 août 2017 09:47 - Frédéric Péters

J'ai rajouté un test pour m'assurer que les payload au nouveau et au format "legacy" sont correctement traités.

Pareil pour l'interrogation du endpoint status.

Je ne vois pas comment la présence du demand_id dans la query string, qui est le format "legacy" est gérée correctement par ce code :

```
@endpoint(perm='can_access', pattern='(?P<demand_id>[\w-]*)')
def status(self, request, demand_id=None, *args, **kwargs):
    """Return demand's status
    """
    if not demand_id and 'demand_id' not in request.GET:
        raise APIError('demand_id is required')

    demand = Demand.objects.get(demand_id=demand_id, resource=self)
```

Et je ne vais pas plus loin, et rembobine jusqu'à ce commentaire que je posais :

Ce serait pas mal d'expliciter un plan dans ce ticket; on passe de "connecteur générique" à "API commune" via un changement de titre du ticket, sans détails, et là ça partirait quand même toutefois vers une part de code commun, avec déjà pour un endpoint un conflit explicite avec l'API existant dans le connecteur MDEL, ce qui amènera une grande joie lorsqu'à la mise à jour de passerelle il faudra en même temps aller éditer des workflows.

Il faut vraiment s'arrêter et définir ce plan, parce que sinon à chaque relecture vont revenir des questions autour; les deux seules voies raisonnables que j'imagine :

- 1) Définir une API commune et rien d'autre, c'est le ticket [#17477](#), on fait gaffe là-bas à ne pas créer d'API incompatible avec le connecteur MDEL existant et on peut rejeter ce ticket.
- 2) Définir une classe de base / mixin contenant les parties spécifiques passerelle d'un connecteur, il faut choisir un connecteur (MDEL ou CityWeb) et adapter/faire celui-ci en mettant en commun du code; c'est galère de gérer deux tickets et donc pareil, je rejetterais celui-ci.

Faire du code "hors sol" ici, ça va continuer à gravement me déranger.

#26 - 11 août 2017 09:50 - Josué Kouka

- Statut changé de En cours à Rejeté

Le plan décidé à la réunion hebdo est de partir sur un connecteur cityweb sans api état civil tout en pensant à le mutualiser avec MDEL in fine. On se concentre donc sur le [#15883](#).

#27 - 14 mars 2018 11:41 - Josué Kouka

- Fichier 0002-ciyweb-use-civil-status-interface-16768.patch ajouté

- Fichier 0001-add-civil-status-connector-interface-16768.patch ajouté

- Statut changé de Rejeté à En cours

Vu qu'il y'a un connecteur AEC à développer pour Dreux, je ré-ouvre ce ticket.
Je pose aussi un patch sur l'interface commune et un second avec le connecteur cityweb utilisant cet api.

#28 - 14 mars 2018 18:56 - Thomas Noël

Ne parlons ici que de 0001, et je relis pas tout le blabla précédent du ticket, mais dans 0001 selon moi il devrait apparaitre l'ensemble des possibilités décrites sur https://dev.entrouvert.org/projects/passerelle/wiki/Connecteur_Etat_Civil au lieu des quelques variables obligatoires. Il manque aussi un try/except sur le json.loads du payload.

(pour 0002 le transforme sur l'ensemble des valeurs, c'est pas bien, effets de bord garantis)

#29 - 19 mars 2018 16:31 - Josué Kouka

Discuté la semaine dernière avec Thomas et Brice.
En plus de l'interface commune, il faudrait fournir ou/des formulaire/s :

- des formulaires w.c.s inspiré de monservicepublic.fr:
 - <https://psl.service-public.fr/mademarche/EtatCivil/demarche?action=NAISSANCE>
 - <https://psl.service-public.fr/mademarche/EtatCivil/demarche?action=MARIAGE>
 - <https://psl.service-public.fr/mademarche/EtatCivil/demarche?action=DECES>
- Un workflow basique envoyant la demande vers un connecteur AEC.

#30 - 22 mars 2018 10:59 - Josué Kouka

Je me disais, est ce qu'il ne serait pas mieux de ne pas fournir des valeurs par défaut pour certains champs. Je parle de champs tel que:

- certificate_type: birth, marriage, death, other
- applicant_status (qualité du demandeur): concerned, partner, parent, child, representative, other
- title: mr, ms, mx

Ces valeurs changent en fonction du connecteur. Vu que le connecteur est censé fournir des sources de données, on peut utiliser les valeurs fournies par le connecteur.

L'avantage que j'y trouve c'est que dans le code du connecteur, on a pas à faire une correspondance entre ces valeurs prédéfinis et les valeurs attendues par l'application à laquelle on se connecte en bout de chaîne.

#31 - 26 mars 2018 14:49 - Josué Kouka

- Fichier 0001-add-civil-status-connector-interface-16768.patch ajouté

Une correction sur le try/except pour le decodage json.
Je ne vois pas trop mettre toute les variables dans le fichier __init__.py du coup, je joins la documentation dans un fichier readme.md.

#32 - 26 mars 2018 22:44 - Benjamin Dauvergne

Josué Kouka a écrit :

Je me disais, est ce qu'il ne serait pas mieux de ne pas fournir des valeurs par défaut pour certains champs. Je parle de champs tel que:

- certificate_type: birth, marriage, death, other
- applicant_status (qualité du demandeur): concerned, partner, parent, child, representative, other
- title: mr, ms, mx

Ces valeurs changent en fonction du connecteur. Vu que le connecteur est censé fournir des sources de données, on peut utiliser les valeurs fournies par le connecteur.

L'avantage que j'y trouve c'est que dans le code du connecteur, on a pas à faire une correspondance entre ces valeurs prédéfinis et les valeurs attendues par l'application à laquelle on se connecte en bout de chaîne.

Oui ça me parait sain que pour les référentiels évidents tu définisses toi même ta nomenclature une fois pour toute et prévoir que chaque implémentation devra fournir un mapping vers sa nomenclature propre, s'agissant de l'état civil ça devrait représenter la totalité.

L'intérêt c'est aussi que ça nous permettra d'avoir aussi un formulaire pure d'état civil pour les villes qui n'ont pas de logiciel ou pas de connecteur.

#33 - 26 mars 2018 22:48 - Benjamin Dauvergne

Josué Kouka a écrit :

Une correction sur le try/except pour le decodage json.
Je ne vois pas trop mettre toute les variables dans le fichier __init__.py du coup, je joins la documentation dans un fichier readme.md.

Je te conseillerai d'éviter de mettre du code dans un *init.py* sauf si quelqu'un t'a donné une bonne raison de le faire, ça évite souvent des problèmes

de boucle d'import, crée plutôt un fichier models.py ou mixin.py.

#34 - 27 mars 2018 19:24 - Josué Kouka

- Fichier *acte-d-etat-civil-msp.wcs* ajouté
- Fichier *demande-d-acte-de-deces-msp.wcs* ajouté
- Fichier *demande-d-acte-de-naissance-msp.wcs* ajouté
- Fichier *demande-d-acte-de-mariage-msp.wcs* ajouté

J'ai attaché les formulaires et le workflow de base.

#35 - 28 mars 2018 09:39 - Frédéric Péters

Adresse électronique (Exemple : nom@exemple.com)

Traditionnellement on utilise le champ "remarque" pour donner les exemples.

#36 - 28 mars 2018 11:45 - Josué Kouka

- Fichier *0001-add-civil-status-connector-interface-16768.patch* ajouté
- Fichier *demande-d-acte-de-naissance-msp.wcs* ajouté
- Fichier *demande-d-acte-de-deces-msp.wcs* ajouté
- Fichier *demande-d-acte-de-mariage-msp.wcs* ajouté

Frédéric Péters a écrit :

Adresse électronique (Exemple : nom@exemple.com)

Traditionnellement on utilise le champ "remarque" pour donner les exemples.

Ok, c'est pris en compte.

Dans le patch je rajoute les datasources par défaut et j'ai pris en compte les remarques de Benjamin.

#37 - 28 mars 2018 16:29 - Josué Kouka

- Fichier *0001-add-civil-status-connector-interface-16768.patch* ajouté

- la réponse du endpoint datasource ne renvoyait pas la clé data, c'est le cas dans ce patch
- je rajoute une description du endpoint datasource.

#38 - 28 mars 2018 22:05 - Benjamin Dauvergne

Maintenant il faut refaire le patch 0002.

#39 - 28 mars 2018 22:06 - Benjamin Dauvergne

Et comme on en est au commentaire 39 je te suggère de pousser une branche, ce sera plus simple à relire.

#40 - 29 mars 2018 15:55 - Josué Kouka

Benjamin Dauvergne a écrit :

Et comme on en est au commentaire 39 je te suggère de pousser une branche, ce sera plus simple à relire.

Ok, voici la branche https://git.entrouvert.org/passerelle.git/log/?h=wip/aec_interface

#41 - 18 avril 2018 15:42 - Brice Mallet

- Fichier *demande-acte-etat-civil-naissance-mariage-ou-deces-generique.wcs* ajouté
- Fichier *transmission-acte-etat-civil-naissance-mariage-ou-deces-generique.wcs* ajouté

Joint :

- le formulaire générique de demande d'état civil (Rq : faudra créer une source de données pour form_var_document_type_withoutfiliation : [{"id": 'without-filiation', 'text': 'Extrait sans filiation'}])
- le workflow générique
Versions en ligne (à Orléans) :
- formulaire = <https://demarches-orleans.test.entrouvert.org/backoffice/forms/82/>
- WF = <https://demarches-orleans.test.entrouvert.org/backoffice/workflows/44/>

#42 - 13 juin 2018 14:34 - Brice Mallet

- Statut changé de *En cours* à *Résolu* (à déployer)

OK pour passer connecteur AEC générique en test en l'état puisque validé par Orléans et Dreux (enfin le sera dès que disponible sur test, on se mord la queue).
Si Stéphane a le temps de valider par ailleurs, tant mieux, sinon je souhaite que l'on bascule en test quand même dès à présent (nécessaire pour Dreux)

#43 - 13 juin 2018 14:53 - Frédéric Péters

Mais dans la branche wip/aec_interface il y a du code à relire. Cette relecture est désormais proposée dans un autre ticket ?

#44 - 13 juin 2018 18:29 - Josué Kouka

Frédéric Péters a écrit :

Mais dans la branche wip/aec_interface il y a du code à relire. Cette relecture est désormais proposée dans un autre ticket ?

Non, c'est au même endroit.

#45 - 13 juin 2018 18:44 - Frédéric Péters

- Statut changé de *Résolu* (à déployer) à *En cours*

La branche est donc à relire.

#46 - 14 juin 2018 09:48 - Frédéric Péters

Vu [#24438](#) j'ai donc aussi regardé la gestion des dates dans le workflow proposé; ça ne va pas.

#47 - 11 septembre 2018 14:55 - Emmanuel Cazenave

Josué qui me demande une relecture par jabber à qui je réponds ici : rebase ta branche.

#48 - 11 septembre 2018 14:56 - Josué Kouka

Emmanuel Cazenave a écrit :

Josué qui me demande une relecture par jabber à qui je réponds ici : rebase ta branche.

Déjà rebasé sur master.

#49 - 11 septembre 2018 16:13 - Emmanuel Cazenave

Josué Kouka a écrit :

Emmanuel Cazenave a écrit :

Josué qui me demande une relecture par jabber à qui je réponds ici : rebase ta branche.

Déjà rebasé sur master.

My bad.

Par contre "grosse branche", déjà d'autres intervenants, je laisse.

#50 - 18 septembre 2018 13:31 - Brice Mallet

Brice Mallet a écrit :

OK pour passer connecteur AEC générique en test en l'état puisque validé par Orléans et Dreux (enfin le sera dès que disponible sur test, on se mord la queue).

Si Stéphane a le temps de valider par ailleurs, tant mieux, sinon je souhaite que l'on bascule en test quand même dès à présent (nécessaire pour Dreux)

Tel qu'indiqué le 13 juin, le formulaire Orléans a été construit pour être très complet, donc adresser toutes les configurations que pourraient se poser, dans le futur, les autres collectivités utilisatrices : en ce sens le formulaire est générique, jusqu'à preuve du contraire.

Mais ce formulaire fait appel à des source de données depuis le connecteur (liste des civilités, des genres, natures d'actes, types de personnes concernées, types de documents) et donc, pour cet aspect, je ne vois pas comment cela pourrait être généraliser puisque les référentiels peuvent être différents de logiciel en logiciel.

Et ce que j'indiquais, pour pouvoir valider dans le cas de / avec Dreux, j'ai besoin de disposer du connecteur en recette pour tester.

#51 - 18 septembre 2018 13:31 - Brice Mallet

- Assigné à changé de Josué Kouka à Serghei Mihai

#52 - 18 septembre 2018 13:38 - Frédéric Péters

Mais ce formulaire fait appel à des source de données depuis le connecteur (liste des civilités, des genres, natures d'actes, types de personnes concernées, types de documents) et donc, pour cet aspect, je ne vois pas comment cela pourrait être généraliser puisque les référentiels peuvent être différents de logiciel en logiciel.

Les référentiels peuvent être différents mais les référentiels sont justement servis par passerelle, quand la donnée revient incorporée à la demande, le connecteur peut l'interpréter.

Imaginons le formulaire, il a un champ genre, alimenté par la source de données "genres" fournie par le connecteur. Le connecteur 1, il annonce m/f, en retour la réponse arrive, elle peut être traitée. Le connecteur 2, il annonce m/f/x, en retour la réponse arrive, elle peut être traitée.

(mais j'accepte l'abandon de l'idée)

#53 - 18 septembre 2018 13:53 - Brice Mallet

(mais j'accepte l'abandon de l'idée)

Et de discuter à l'instant avec Serghei, et vu les échéances pour Dreux (mise en production début octobre alors que chantier ouvert depuis mars sur ce projet particulier), je suis pour malheureusement encore louper cette occasion et produire un 3ème connecteur état civil, celui pour Mélodie/ActesWeb d'Arpège.

#54 - 28 septembre 2018 17:19 - Brice Mallet

- Statut changé de En cours à Rejeté

Le connecteur générique attendra donc, malheureusement, encore... la prochaine demande d'un client avec un autre logiciel ou du temps pour génériciser l'existant

Fichiers

0001-add-civil_status-generic-API-interface-16768.patch	1,63 ko	18 juin 2017	Serghei Mihai
0001-add-civil-status-common-api-16768.patch	5,16 ko	07 juillet 2017	Josué Kouka
0001-add-civil-status-common-api-16768.patch	4,88 ko	07 juillet 2017	Josué Kouka
0001-add-civil-status-common-api-16768.patch	4,71 ko	07 juillet 2017	Josué Kouka
0001-add-civil-status-common-api-16768.patch	5,7 ko	27 juillet 2017	Josué Kouka
0001-add-civil-status-common-api-16768.patch	6,6 ko	01 août 2017	Josué Kouka
0002-make-MDEL-CivilStatusMixin-compliant-16768.patch	10,8 ko	07 août 2017	Josué Kouka
0001-add-civil-status-common-api-16768.patch	6,48 ko	07 août 2017	Josué Kouka
0002-ciyweb-use-civil-status-interface-16768.patch	7,67 ko	14 mars 2018	Josué Kouka
0001-add-civil-status-connector-interface-16768.patch	2,94 ko	14 mars 2018	Josué Kouka
0001-add-civil-status-connector-interface-16768.patch	13,6 ko	26 mars 2018	Josué Kouka
demande-d-acte-de-deces-msp.wcs	10,4 ko	27 mars 2018	Josué Kouka
acte-d-etat-civil-msp.wcs	15,1 ko	27 mars 2018	Josué Kouka
demande-d-acte-de-naissance-msp.wcs	13,1 ko	27 mars 2018	Josué Kouka
demande-d-acte-de-mariage-msp.wcs	12,7 ko	27 mars 2018	Josué Kouka
0001-add-civil-status-connector-interface-16768.patch	16,4 ko	28 mars 2018	Josué Kouka

demande-d-acte-de-naissance-msp.wcs	13,1 ko	28 mars 2018	Josué Kouka
demande-d-acte-de-deces-msp.wcs	10,5 ko	28 mars 2018	Josué Kouka
demande-d-acte-de-mariage-msp.wcs	12,7 ko	28 mars 2018	Josué Kouka
0001-add-civil-status-connector-interface-16768.patch	16,3 ko	28 mars 2018	Josué Kouka
demande-acte-etat-civil-naissance-mariage-ou-deces-generique.wcs	21,3 ko	18 avril 2018	Brice Mallet
transmission-acte-etat-civil-naissance-mariage-ou-deces-generique.wcs	6,2 ko	18 avril 2018	Brice Mallet