

## Passerelle - Development #8385

### teamnet\_axel: une facture de l'historique ne devrait pas être payable en ligne

25 septembre 2015 17:46 - Serghei Mihai (conгés, retour 15/05)

<b>Statut:</b>	Fermé	<b>Début:</b>	25 septembre 2015
<b>Priorité:</b>	Normal	<b>Echéance:</b>	
<b>Assigné à:</b>	Serghei Mihai (conгés, retour 15/05)	<b>% réalisé:</b>	0%
<b>Catégorie:</b>		<b>Temps estimé:</b>	0:00 heure
<b>Version cible:</b>		<b>Planning:</b>	
<b>Patch proposed:</b>	Oui		

**Description**

Actuellement une facture est payable en ligne si son montant dû est non-nul et la date d'échéance n'est pas dépassée. Les factures en prélevement apparaissent dans l'historique des factures mais ne respectent pas ses conditions: donc considérées comme à payer.

Il faudrait, donc, vérifier que la facture ne fait pas partie des factures de l'historique

#### Révisions associées

##### Révision 71ac21bf - 07 octobre 2015 11:43 - Serghei Mihai (conгés, retour 15/05)

contrib.teamnet\_axel: code refactoring (#8385)

Move all invoices retrieval logic into the model.

##### Révision a139c438 - 07 octobre 2015 11:43 - Serghei Mihai (conгés, retour 15/05)

contrib.teamnet\_axel: mark historical invoices as not payable (#8385)

##### Révision af54e563 - 07 octobre 2015 18:39 - Serghei Mihai (conгés, retour 15/05)

contrib.teamnet\_axel: fix invoice payment notification (#8385)

#### Historique

##### #1 - 25 septembre 2015 18:46 - Serghei Mihai (conгés, retour 15/05)

- Fichier 0001-contrib.teamnet\_axel-fix-invoice-online-payment-cond.patch ajouté
- Fichier 0002-contrib.teamnet\_axel-forbid-online\_payment-of-histor.patch ajouté
- Statut changé de Nouveau à En cours
- Patch proposed changé de Non à Oui

##### #2 - 25 septembre 2015 18:58 - Frédéric Péters

Ça va devenir vraiment difficile à suivre, on a un get\_invoices() qui appelle un service différent selon la présence ou non d'un paramètre months, et après on redivise le résultat.

Est-ce que du coup on ne pourrait pas tout faire dans le get\_invoices() et obtenir quelque chose de plus clair ?

##### #3 - 25 septembre 2015 19:03 - Frédéric Péters

Quelque chose dans ce genre :

```
# teamnet split of invoices doesn't suit us, get them all and do the
# split on our own terms.
operation = 'FacturesApayerRegie'
xml_invoices = ET.Element('LISTFACTURE')
ET.SubElement(xml_invoices, 'IDREGIE').text = regie_id
ET.SubElement(xml_invoices, 'IDFAMILLE').text = family_id
data = self.get_data(operation, xml_invoices)
payable_invoices = data.findall('PORTAIL/FACTURES')
payable_invoices = [utils.normalize_invoice(i.attrib, family_id) for i in payable_invoices]
```

```
operation = 'HistoriqueFacturesRegie'
ET.SubElement(xml_invoices, 'NBMOIS').text = '12'
data = self.get_data(operation, xml_invoices)
```

```
invoices_history = data.findall('PORTAIL/FACTURES')
invoices_history = [utils.normalize_invoice(i.attrib, family_id,
    online_payment=False) for i in invoices_history]
```

```
# remove duplicates
all_invoices = invoices_history
for invoice in payable_invoices:
    if not invoice in all_invoices: # exemple, ça doit fonctionner sur le numéro de facture
        all_invoices.append(invoice)
```

```
return sorted(all_invoices, key=lambda i: i['created'], reverse=True)
```

#### #4 - 29 septembre 2015 13:47 - Serghei Mihai (congé, retour 15/05)

- Fichier 0001-contrib.teamnet\_axel-forbid-online\_payment-of-histor.patch ajouté

Ok pour que tout se fasse dans get\_invoices, avec séparation des factures à payer et l'historique.

#### #5 - 29 septembre 2015 14:09 - Serghei Mihai (congé, retour 15/05)

- Fichier 0001-contrib.teamnet\_axel-forbid-online\_payment-of-histor.patch supprimé

#### #6 - 29 septembre 2015 14:21 - Serghei Mihai (congé, retour 15/05)

- Fichier 0001-contrib.teamnet\_axel-forbid-online\_payment-of-histor.patch ajouté

#### #7 - 04 octobre 2015 12:24 - Serghei Mihai (congé, retour 15/05)

Si quelqu'un peut commenter ce patch, je suis preneur

#### #8 - 04 octobre 2015 21:51 - Benjamin Dauvergne

Je ne comprends pas bien la description du ticket, mais il me semble que ce qui est dit c'est que les factures en prélèvement apparaissent dans les factures à payer alors qu'elles ne devraient pas et donc on doit vérifier qu'elles ne sont pas aussi dans l'historique. Et là ce qui est fait c'est l'inverse, on supprime de l'historique les factures qui apparaissent aussi dans la liste des factures à payer, soit c'est mal décrit soit c'est mal compris mais il y a un truc.

Ensuite tu n'as pas suivi un commentaire de Fred dans son exemple de code:

```
if not invoice in all_invoices: # exemple, ça doit fonctionner sur le numéro de facture
```

ça ne me paraît pas sain de comparer les factures directement via les structures, on ne sait jamais ce qui sera renvoyé, il vaut mieux se baser uniquement sur le invoice['id'], je dirai que le plus simple ce serait d'utiliser des dictionnaires plutôt que des listes dans une bonne partie du code. On pourrait faire l'effort d'avoir deux méthodes, une get\_payable\_invoices() et une autre get\_historical\_invoices() renvoyant chacune un dico indexé par les invoice['id'].

À mon avis des tests avec python-mock pour simuler les retours de get\_\*\_invoices seraient utiles.

#### #9 - 04 octobre 2015 22:59 - Serghei Mihai (congé, retour 15/05)

Benjamin Dauvergne a écrit :

Je ne comprends pas bien la description du ticket, mais il me semble que ce qui est dit c'est que les factures en prélèvement apparaissent dans les factures à payer alors qu'elles ne devraient pas et donc on doit vérifier qu'elles ne sont pas aussi dans l'historique. Et là ce qui est fait c'est l'inverse, on supprime de l'historique les factures qui apparaissent aussi dans la liste des factures à payer, soit c'est mal décrit soit c'est mal compris mais il y a un truc.

Initialement les factures en prélèvement apparaissaient dans les factures à payer et donc je ne faisais aucune différence entre facture à payer("normale") ou en prélèvement. Cela a été validé avec les gens de Fontenay.

Suite aux changements dans leur base, ou je ne sais pas quoi, les factures en prélèvement apparaissent désormais dans l'historique et ne doivent pas être payables en ligne.

Donc je marque toutes les factures de l'historique comme non-payables.

Ensuite tu n'as pas suivi un commentaire de Fred dans son exemple de code:

[...]

```
operation = 'HistoriqueFacturesRegie'
ET.SubElement(xml_invoices, 'NBMOIS').text = '12'
data = self.get_data(operation, xml_invoices)
```

Cet appel renvoie TOUTES les factures: celles à payer et celles de l'historique. Et donc j'exclus de cette liste la liste des factures à payer pour obtenir

le vrai historique.

Or dans la suggestion de Fred je vois l'aggrégation des factures à payer avec celles de l'historique.

ça ne me paraît pas sain de comparer les factures directement via les structures, on ne sait jamais ce qui sera renvoyé, il vaut mieux se baser uniquement sur le `invoice['id']`, je dirai que le plus simple ce serait d'utiliser des dictionnaires plutôt que des listes dans une bonne partie du code.

Nous avons défini que les factures étaient renvoyées sous forme d'une liste de dicos:

[https://dev.entrouvert.org/projects/passerelle/wiki/Connecteur\\_famille#Gestionremont%C3%A9-des-factures](https://dev.entrouvert.org/projects/passerelle/wiki/Connecteur_famille#Gestionremont%C3%A9-des-factures) et donc comme le résultat est déjà une liste, je préférerais de ne pas inventer d'autres structures des données (même si plus simples).

On pourrait faire l'effort d'avoir deux méthodes, une `get_payable_invoices()` et une autre `get_historical_invoices()` renvoyant chacune un dico indexé par les `invoice['id']`.

Le seul intérêt que je vois d'avoir 2 méthodes séparées est que `get_payable_invoices` appellerait un seul webservice ("FacturesApayerRegie"). Pour `get_historical_invoices` c'est toujours 2 appels ("HistoriqueFacturesRegie" et "FacturesApayerRegie")

#### #10 - 04 octobre 2015 23:35 - Benjamin Dauvergne

Serghei Mihai a écrit :

Benjamin Dauvergne a écrit :

Je ne comprends pas bien la description du ticket, mais il me semble que ce qui est dit c'est que les factures en prélèvement apparaissent dans les factures à payer alors qu'elles ne devraient pas et donc on doit vérifier qu'elles ne sont pas aussi dans l'historique. Et là ce qui est fait c'est l'inverse, on supprime de l'historique les factures qui apparaissent aussi dans la liste des factures à payer, soit c'est mal décrit soit c'est mal compris mais il y a un truc.

Initialement les factures en prélèvement apparaissaient dans les factures à payer et donc je ne faisais aucune différence entre facture à payer ("normale") ou en prélèvement. Cela a été validé avec les gens de Fontenay.

Suite aux changements dans leur base, ou je ne sais pas quoi, les factures en prélèvement apparaissent désormais dans l'historique et ne doivent pas être payables en ligne.

Donc je marque toutes les factures de l'historique comme non-payables.

Ok mais ça ne correspond pas à la description du ticket (mais le titre est ok). Si tu peux la relire et l'éditer c'est cool. Il me semble qu'ils ont juste rendu leurs web-services logiques dans ce cas, le web-service à payer renvoie les factures à payer, modulo des histoires de montant supérieur à zéro et de date limite de paiement dans le futur; l'autre web-service renvoie tout, c'est logique.

Ensuite tu n'as pas suivi un commentaire de Fred dans son exemple de code:

[...]

[...]

Cet appel renvoie TOUTES les factures: celles à payer et celles de l'historique. Et donc j'exclus de cette liste la liste des factures à payer pour obtenir le vrai historique.

Or dans la suggestion de Fred je vois l'aggrégation des factures à payer avec celles de l'historique.

La suggestion c'est de comparer les factures par leur 'id' je n'ai pas dit de reprendre son code.

ça ne me paraît pas sain de comparer les factures directement via les structures, on ne sait jamais ce qui sera renvoyé, il vaut mieux se baser uniquement sur le `invoice['id']`, je dirai que le plus simple ce serait d'utiliser des dictionnaires plutôt que des listes dans une bonne partie du code.

Nous avons défini que les factures étaient renvoyées sous forme d'une liste de dicos:

[https://dev.entrouvert.org/projects/passerelle/wiki/Connecteur\\_famille#Gestionremont%C3%A9-des-factures](https://dev.entrouvert.org/projects/passerelle/wiki/Connecteur_famille#Gestionremont%C3%A9-des-factures) et donc comme le résultat est déjà une liste, je préférerais de ne pas inventer d'autres structures des données (même si plus simples).

Je n'ai pas dit de changer les structures de données renvoyées par `get_invoices()` seulement de s'appuyer sur deux méthodes annexes qui renverraient des dicos, ensuite tu peux très bien reconstruire des listes.

On pourrait faire l'effort d'avoir deux méthodes, une `get_payable_invoices()` et une autre `get_historical_invoices()` renvoyant chacune un dico indexé par les `invoice['id']`.

Le seul intérêt que je vois d'avoir 2 méthodes séparées est que `get_payable_invoices` appellerait un seul webservice ("FacturesApayerRegie").

Pour `get_historical_invoices` c'est toujours 2 appels ("HistoriqueFacturesRegie" et "FacturesApayerRegie")

Non, chaque méthode n'appellerait qu'un web-service l'idée étant d'extraire la partie interrogation des web-services Teamnet histoire de rendre le tout lisible et testable, un jour.

Au final on aurait 5 méthodes:

- `get_teamnet_payable_invoices()` -> renvoie un dict(id -> facture), ici sont normalisées les factures
- `get_teamnet_historical_invoices()` -> renvoie un dict(id -> facture), ici sont normalisées les factures, mais en plus elles sont toutes marquées non payables
- `get_payable_invoices()` -> appelle seulement `get_teamnet_payable_invoices()` et renvoie une liste
- `get_historical_invoices()` -> appelle `get_teamnet_payable_invoices()` et `get_teamnet_historical_invoices()` et renvoie une liste
- `get_invoice()` -> appelle `get_teamnet_payable_invoices()` regarde si la facture à trouver est dedans, sinon appelle `get_teamnet_historical_invoices()`

On a inventé une deuxième notion d'historique alors qu'il y a déjà celle de teamnet, je suppose qu'il y a une raison fonctionnelle du côté portail mais il faut que cela se voit dans le code. Avec une docstring sur `get_historical_invoices()` qui explique pourquoi on soustrait le résultat d'une méthode à l'autre et ce sera parfait.

#### #11 - 07 octobre 2015 11:08 - Serghei Mihai (congé, retour 15/05)

- Fichier `0001-contrib.teamnet_axel-code-refactoring-8385.patch` ajouté

- Fichier `0002-contrib.teamnet_axel-mark-historical-invoices-as-not.patch` ajouté

Code refactorisé pour avoir les 5 méthodes et toute la logique des gestion des factures dans le modèle. Ensuite je marque les factures de l'historique comme non-payables en ligne.

Testé chaque méthode avec les familles de test fournies par Fontenay

#### #12 - 07 octobre 2015 11:29 - Frédéric Péters

À l'occasion (eoday), il faudra que Benjamin te montre l'utilisation de mock.

```
adults = [normalize_person(i) for i in individus if i['indtype'] in ('1', '2')]
children = [normalize_person(i) for i in individus if i['indtype'] == '3']
```

On devine ce que sont '1', '2' et '3', mais c'est le genre de truc qui pourrait être plus clair avec des constantes `ADULTE1 = '1'`, etc.

#### #13 - 07 octobre 2015 11:44 - Serghei Mihai (congé, retour 15/05)

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

J'ai rajouté les constantes et j'ai poussé:

```
commit a139c438b22befd8e4cb90763b1119842d2e39a2
Author: Serghei Mihai <smihai@entrouvert.com>
Date: Tue Oct 6 17:55:56 2015 +0200
```

```
contrib.teamnet_axel: mark historical invoices as not payable (#8385)
```

```
commit 71ac21bf746ableaff94fa6acbalc0947ad994aa
Author: Serghei Mihai <smihai@entrouvert.com>
Date: Tue Oct 6 13:00:03 2015 +0200
```

```
contrib.teamnet_axel: code refactoring (#8385)
```

```
Move all invoices retrieval logic into the model.
```

#### #14 - 27 mars 2017 11:02 - Serghei Mihai (congé, retour 15/05)

- Statut changé de *Résolu* (à déployer) à *Fermé*

### Fichiers

<code>0001-contrib.teamnet_axel-fix-invoice-online-payment-cond.patch</code>	1,09 ko 25 septembre 2015	Serghei Mihai (congé, retour 15/05)
<code>0002-contrib.teamnet_axel-forbid-online_payment-of-histor.patch</code>	3,71 ko 25 septembre 2015	Serghei Mihai (congé, retour 15/05)
<code>0001-contrib.teamnet_axel-forbid-online_payment-of-histor.patch</code>	5,6 ko 29 septembre 2015	Serghei Mihai (congé, retour 15/05)
<code>0002-contrib.teamnet_axel-mark-historical-invoices-as-not.patch</code>	2,2 ko 07 octobre 2015	Serghei Mihai (congé, retour 15/05)
<code>0001-contrib.teamnet_axel-code-refactoring-8385.patch</code>	10,4 ko 07 octobre 2015	Serghei Mihai (congé, retour 15/05)