

## w.c.s. - Development #12916

### appel webservice : gestion du code "err" et d'un header d'erreur

24 août 2016 11:58 - Thomas Noël

<b>Statut:</b>	Fermé	<b>Début:</b>	24 août 2016
<b>Priorité:</b>	Normal	<b>Echéance:</b>	
<b>Assigné à:</b>	Thomas Noël	<b>% réalisé:</b>	0%
<b>Catégorie:</b>		<b>Temps estimé:</b>	0:00 heure
<b>Version cible:</b>	v1.61	<b>Planning:</b>	
<b>Patch proposed:</b>	Oui		
<b>Description</b>			
Par convention, Passerelle renvoie toujours un 200 quand elle a réussi à contacter le système tiers. Si ce système répond une erreur, celle-ci se retrouve traduite dans le JSON de retour :			
<pre>{   data: ...   err: 0 ou non-0 }</pre>			
En retour du webservice, on devrait pouvoir programmer un saut si « err!=0 », au même niveau que les saut en cas de 4xx, 5xx, etc.			
Par ailleurs, dans le cas d'un retour non-json (document), on détectera si un header « X-Passerelle-Return-Err » existe et est !=0. Dans ce cas, on considère aussi qu'il y a une erreur en retour (et on stocke le code dans wsvarname_return_err).			
<b>Demandes liées:</b>			
Lié à Passerelle - Development #12924: Ajout header http dans décorateur json		<b>Fermé</b>	<b>24 août 2016</b>

#### Révisions associées

##### Révision 9fc58ea7 - 03 octobre 2016 17:53 - Thomas Noël

wscall: add action if error in application (#12916)

#### Historique

##### #1 - 24 août 2016 14:43 - Thomas Noël

- Sujet changé de gestion du code "err" dans le retour d'un appel de webservice à appel webservice : gestion du code "err" et d'un header d'erreur
- Description mis à jour

##### #2 - 24 août 2016 14:51 - Thomas Noël

- Lié à Development #12924: Ajout header http dans décorateur json ajouté

##### #3 - 24 août 2016 15:28 - Thomas Noël

- Assigné à mis à Thomas Noël

##### #4 - 24 août 2016 17:36 - Thomas Noël

- Fichier 0001-wscall-add-action-if-err-in-json-data-or-specific-he.patch ajouté
- Statut changé de Nouveau à En cours
- Patch proposed changé de Non à Oui

##### #5 - 26 août 2016 15:02 - Frédéric Péters

Peut-être ne pas avoir "passerelle" dans le nom de l'entête ?

Mettre les tests sur la gestion du code "err" dans un test séparé.

S'il n'y a pas d'entête et qu'on ne définit pas de variable, il faut quand même que la gestion d'erreur fonctionne. (déléguer un bout dans le store\_response n'est pas correct)

Concrètement, ce test ne passe pas :

```
item = WebserviceCallStatusItem()
item.url = 'http://remote.example.net/json-err1'
item.post = False
item.action_on_error_code = ':stop'
with pytest.raises(AbortActionException):
    item.perform(formdata)
```

#### #6 - 06 septembre 2016 15:14 - Thomas Noël

Thanks, tout a fait exact pour le store\_response.

Voici une nouvelle mouture du patch. A noter que je ne suis pas à l'aise avec l'intitulé «Action on JSON or header error» que j'aimerais traduire en français par «Action en cas de retour d'erreur sur le tiers» ou «Action en cas d'erreur applicative» (pour tenter de faire une différence avec les autres, qui sont plus des "crashes" ou des "erreurs anormales")

#### #7 - 06 septembre 2016 15:14 - Thomas Noël

- Fichier 0001-wscall-add-action-if-err-in-json-data-or-specific-he.patch ajouté

#### #8 - 16 septembre 2016 10:17 - Frédéric Péters

Simplement "Action on application error" ?

#### #9 - 19 septembre 2016 14:20 - Thomas Noël

- Fichier 0001-wscall-add-action-if-error-in-application-12916.patch ajouté

Yep, cool, j'en profite pour :

- passer cette action en premier dans la liste, car ça devient finalement la plus pertinente (celle qui a le plus de sens dans un workflow)
- renommer les error\_\* en app\_error\_\*

#### #10 - 03 octobre 2016 10:19 - Frédéric Péters

Pour la couverture, il faudrait ajouter un test sur x-error-code pas un entier et sur l'attente de json mais la réception de xml; en gros pour tester les deux except: ajoutés ici :

```
+         # result is good only if header value is '0'
+         try:
+             app_error_code = int(app_error_code_header)
+         except ValueError as e:
+             app_error_code = app_error_code_header
+     elif self.response_type == 'json':
+         try:
+             d = json_loads(data)
+         except (ValueError, TypeError) as e:
+             pass
```

#### #11 - 03 octobre 2016 12:52 - Thomas Noël

- Fichier 0001-wscall-add-action-if-error-in-application-12916.patch ajouté

- Fichier juste-pour-info-le-patch-du-patch.patch ajouté

Voici, j'ai en profité pour tester aussi d'autres cas (retour d'un json en tant qu'attachement, où le 'err' ne doit donc pas être considéré comme un erreur, sauf s'il est en header)

#### #12 - 03 octobre 2016 13:24 - Frédéric Péters

```
item = WebserviceCallStatusItem()
item.url = 'http://remote.example.net/xml-errheader'
item.post = False
item.varname = 'xxx'
item.response_type = 'json' # wait for json but receive xml
item.record_errors = True
```

Mais ça teste la situation où il y a un entête, je pensais à un test où on reçoit du xml alors qu'on attend du json, et qu'il n'y a pas d'entête. (item.url = 'http://remote.example.net/xml', item.response\_type = 'json').

#### #13 - 03 octobre 2016 15:17 - Thomas Noël

Frédéric Péters a écrit :

Mais ça teste la situation où il y a un entête, je pensais à un test où on reçoit du xml alors qu'on attend du json, et qu'il n'y a pas d'entête. (item.url = 'http://remote.example.net/xml', item.response\_type = 'json').

Je pense que c'est déjà testé dans « test\_webservice\_call » :

```
812     item = WebserviceCallStatusItem()
813     item.url = 'http://remote.example.net/xml'
814     item.post = False
815     item.perform(formdata)
```

A noter qu'on considère que "ça passe", car il n'y a pas de retour d'erreur explicite... juste c'est pas du json, mais c'est un 200 et il n'y a ni "err" (pas de json) ni entête... je ne sais pas vraiment quoi faire, vrai dire : doit-on considérer que c'est un cas de "app\_error" ?

#### #14 - 03 octobre 2016 15:50 - Frédéric Péters

Oui, davantage testé, pour la gestion des erreurs, un peu plus loin, via :

```
item.action_on_bad_data = ':stop'
with pytest.raises(AbortActionException):
    item.perform(formdata)
```

Et pour moi cette gestion d'erreur est ok.

Donc je dirais ack au patch.

#### #15 - 03 octobre 2016 17:54 - Thomas Noël

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

```
commit 9fc58ea7db4fd871aef0ed40890d43cf53b25ea2
Author: Thomas NOEL <tnoel@entrouvert.com>
Date:   Wed Aug 24 17:34:01 2016 +0200
```

```
wscall: add action if error in application (#12916)
```

#### #16 - 04 décembre 2016 13:01 - Frédéric Péters

- Version cible mis à *v1.61*

#### #17 - 04 décembre 2016 13:12 - Frédéric Péters

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

### Fichiers

0001-wscall-add-action-if-err-in-json-data-or-specific-he.patch	11,2 ko	24 août 2016	Thomas Noël
0001-wscall-add-action-if-err-in-json-data-or-specific-he.patch	13 ko	06 septembre 2016	Thomas Noël
0001-wscall-add-action-if-error-in-application-12916.patch	13,1 ko	19 septembre 2016	Thomas Noël
0001-wscall-add-action-if-error-in-application-12916.patch	16,1 ko	03 octobre 2016	Thomas Noël
juste-pour-info-le-patch-du-patch.patch	5,89 ko	03 octobre 2016	Thomas Noël