

w.c.s. - Development #39030

conséquence de l'évaluation "lazy" des champs wcs

16 janvier 2020 17:09 - Serghei Mihai

Statut:	Fermé	Début:	16 janvier 2020
Priorité:	Normal	Echéance:	
Assigné à:	Frédéric Péters	% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	Non
Patch proposed:	Oui		

Description

Suite au passage à l'évaluation "lazy" des champs certains formulaires et workflows peuvent rencontrer le souci signalé dans #38960: si le formulaire comporte les variables form_var_numero et form_var_numero_service, l'évaluation du champ form_var_numero_service est en fait l'évaluation d'un attribut service qui n'existe pas du champ form_var_numero.

Il risque d'y avoir pas mal de formulaires et workflows en production dans ce genre de situation et comme dit un CPF lyonnais:

sans vouloir en faire trop.. y'a moyen que ça mette un peu un gros bordel

Révisions associées

Révision dbe8ba56 - 03 février 2020 11:31 - Serghei Mihai

workflows: clean context after workflow form action (#39030)

Révision 3d9fd7f2 - 06 février 2020 11:07 - Serghei Mihai

workflows: clean context after workflow form action (#39030)

Historique

#1 - 16 janvier 2020 17:19 - Stéphane Laget

- Description mis à jour

#2 - 16 janvier 2020 17:58 - Frédéric Péters

- Tracker changé de Support à Development
- Projet changé de Publik à w.c.s.
- Statut changé de Nouveau à Information nécessaire
- Assigné à mis à Serghei Mihai

'évaluation du champ form_var_numero_service est en fait l'évaluation d'un attribut service qui n'existe pas du champ form_var_numero.

Ça ne devrait pas être le cas et il y a des tests pour vérifier ce cas :

```
formdef.name = 'foobarlazy'  
...  
formdef.fields = [  
  fields.StringField(id='0', label='string', varname='foo_foo'),  
  ...  
  fields.StringField(id='6', label='string2', varname='foo_foo_baz_baz'),  
  ...  
  assert context['form_var_foo_foo'] == 'bar'  
  ...  
  assert context['form_var_foo_foo_baz_baz'] == 'other'  
  ...  
]
```

#3 - 16 janvier 2020 17:59 - Frédéric Péters

(pour référence on a l'évaluation lazy sur les conditions de champs depuis qu'elles existent, quand même).

#4 - 17 janvier 2020 10:34 - Benjamin Dauvergne

- *Projet changé de w.c.s. à Publik*
- *Assigné à changé de Serghei Mihai à Benjamin Dauvergne*
- *Club mis à Non*

On peut fermer ce ticket où quelqu'un a vraiment constaté ce comportement ?

#5 - 17 janvier 2020 10:34 - Benjamin Dauvergne

- *Projet changé de Publik à w.c.s.*
- *Assigné à changé de Benjamin Dauvergne à Serghei Mihai*

#6 - 27 janvier 2020 16:51 - Serghei Mihai

- *Statut changé de Information nécessaire à En cours*

Même souci signalé dans #38921: variable de formulaire form_var_email et variable de workflow form_var_email_destinataire. Dans la condition d'exécution d'une action form_var_email_destinataire est évalué à None.

La différence par rapport au contenu du test mentionné plus haut est qu'une est une variable de formulaire et l'autre de workflow.

J'essaie de reproduire le problème.

#7 - 27 janvier 2020 20:18 - Serghei Mihai

- *Fichier 0001-tests-test-lazy-workflow-data.patch ajouté*
- *Patch proposed changé de Non à Oui*

Réproduit dans ce test. Je me penche sur un correctif.

#8 - 27 janvier 2020 20:48 - Frédéric Péters

Histoire de limiter les changements, il me semble qu'on peut :

```
@@ -580,14 +580,6 @@ def variable_test_data(pub):
     fields.StringField(id='1', label='string', varname='foo_foo_baz'),
 ]
     st_status = workflow.add_status('New', 'new')
-     jump = JumpWorkflowStatusItem()
-     jump.id = '_jump'
-     jump.by = ['_submitter', '_receiver']
-     jump.status = 'finished'
-     jump.parent = st_status
-
-     end_status = workflow.add_status('Finished', 'finished')
-     st_status.items.append(jump)
     workflow.store()

     FormDef.wipe()
```

#9 - 27 janvier 2020 21:06 - Frédéric Péters

Mais non le test.

```
     workflow.fields = [
         fields.StringField(id='1', label='string', varname='foo_foo_baz'),
     ]
```

Ça ne veut rien dire, un workflow n'a pas d'attribut "fields", il peut avoir des champs sous variables_formdef (les options de workflow) ou backoffice_fields_formdef (les données de traitement).

#10 - 27 janvier 2020 21:08 - Frédéric Péters

Et donc, à faire :

```
     workflow = Workflow(name='bar')
-     workflow.fields = [
-         fields.StringField(id='1', label='string', varname='foo_foo_baz'),
+     workflow.backoffice_fields_formdef = WorkflowBackofficeFieldsFormDef(workflow)
+     workflow.backoffice_fields_formdef.fields = [
+         fields.StringField(id='bol', label='string', varname='foo_foo_baz'),
```

```
]
st_status = workflow.add_status('New', 'new')
```

#11 - 27 janvier 2020 21:13 - Frédéric Péters

Et plus loin, je ne sais pas ce qui est recherché dans :

```
- workflow.data = {
-   '1': 'workflow baz'
- }
```

mais les données de traitement vont dans formdata.data, ce qui serait donc

```
+ formdata.data['bol'] = 'workflow baz'
+ formdata.store()
```

et avec ça on obtient la valeur attendue lors de

```
+ assert context['form_var_foo_foo_baz'] == 'workflow baz'
```

Voilà donc, je ne sais pas comment l'analyse de la situation a procédé mais le test derrière ne correspond pas à une réalité.

#13 - 29 janvier 2020 00:19 - Serghei Mihai

- Fichier *form-test-donnees-de-traitement.wcs* ajouté

- Fichier *workflow-wf-demande-renseignement-voirie-pour-debug.wcs* ajouté

Formulaire et workflow permettant de reproduire le problème.

Dans le statut "Choix du service destinataire", une fois un mail alternatif choisi il doit être affiché dans l'historique lors du passage dans le statut "Transmission de la demande".

#15 - 30 janvier 2020 11:53 - Stéphane Laget

- Description mis à jour

#16 - 30 janvier 2020 14:01 - Serghei Mihai

- Fichier *0001-tests-test-lazy-eval-from-multiple-sources-39030.patch* ajouté

- Description mis à jour

Voici le test qui représente le cas qui arrive dans le workflow attaché: la source `wcs.fields.ConditionVars` rajoute les variables du formulaire dans le contexte comme `form_var_email` et c'est `form_var_email_destinataire` qui y est recherché.

#17 - 30 janvier 2020 17:32 - Serghei Mihai

Je sèche sur l'algo de recherche de la bonne valeur solution.

S'il y a des cerveaux plus rapides que le mien, je ne suis que preneur.

#18 - 31 janvier 2020 11:14 - Frédéric Péters

Il y a moyen de commencer par un test qui corresponde à du vrai code exécuté, plutôt qu'un truc dont l'artificialité éloigne la réflexion ? (et je regarderai)

#19 - 02 février 2020 20:27 - Serghei Mihai

- Fichier *0001-tests-test-lazy-eval-from-multiple-sources-39030.patch* ajouté

- Statut changé de *En cours* à *Solution proposée*

Ok. Test avec action en backoffice qui provoque l'échec de calcul de la variable de traitement ayant le même espace de nommage qu'une variable du formulaire.

#20 - 02 février 2020 22:55 - Benjamin Dauvergne

C'est très frustrant pour moi qu'on ne pousse pas une branche dans ce genre de cas, ça permet immédiatement de voir le problème en action sans avoir à appliquer le patch localement.

#21 - 02 février 2020 23:14 - Frédéric Péters

- Fichier *0001-workflows-clean-context-after-workflow-form-action-3.patch* ajouté

Mais, pour continuer à essayer de comprendre les éléments en jeu; si je fais ça sur le test :

```
formdef.fields = [
-     fields.StringField(id='0', label='Foo', varname='foo')
+     fields.StringField(id='0', label='Foo', varname='whatever')
]
```

toutes les explications passées, de confusion entre le champ du formulaire et la donnée de traitement, me laisseraient penser que ça devrait résoudre l'affaire. Et pourtant le test échoue quand même.

Je continue de déconstruire le test, comprendre les bouts, genre pourquoi donc un firstof, et à force de déconstruire,

```
- st2.items.append(jump)
  st2.items.append(setbo)
+ st2.items.append(jump)
```

que la donnée de traitement soit posée avant le saut, vu que le saut est basé sur la condition.

Et enfin, à ce moment, le test semble rejoindre ce qui est cherché.

De là, répondre que non le firstof n'a pas de rapport,

```
- setbo.fields = [{'field_id': 'bo1', 'value': '% firstof local_var_str2 local_var_str1 %'}]
+ setbo.fields = [{'field_id': 'bo1', 'value': 'go'}]
```

et en arriver juste à faire comme à l'édition d'une demande, l'appel à `clean_submission_context()`,

```
@@ -216,6 +216,7 @@ class FormWorkflowStatusItem(WorkflowStatusItem):
     if form.get_submit() == 'submit' and not form.has_errors():
         self.evaluate_live_form(form, formdata, user)
         formdata.store()
+     get_publisher().substitutions.unfeed(lambda x: x.__class__.__name__ == 'ConditionVars')

def get_parameters_view(self):
```

#24 - 02 février 2020 23:27 - Benjamin Dauvergne

J'ai lié aux tickets #37017 et #37093 parce qu'on a jamais trouvé la source du problème et j'aimerais comprendre.

#25 - 03 février 2020 09:45 - Serghei Mihai

- Statut changé de Solution proposée à Solution validée

- Assigné à changé de Serghei Mihai à Frédéric Péters

Frédéric Péters a écrit :

[...]

que la donnée de traitement soit posée avant le saut, vu que le saut est basé sur la condition.

Effectivement je me suis melé les pinceaux dans l'ordre de déclaration des actions. Et le firstof était pour me rapprocher du workflow joint plus haut. Une simple affectation suffisait.

et en arriver juste à faire comme à l'édition d'une demande, l'appel à `clean_submission_context()`,

Ma première idée fut de trouver une solution au niveau de l'algo de recherche du champ dans le contexte alimenté par toutes les sources. Mais seule `ConditionsVars` empêche de remonter jusqu'à `form` dans le contexte et ta solution me paraît bien.

#26 - 03 février 2020 11:32 - Frédéric Péters

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

```
commit dbe8ba569ca397a6d4d9996b0a3f07631a3ba4db
Author: Serghei Mihai <smihai@entrouvert.com>
Date: Mon Jan 27 20:10:03 2020 +0100
```

```
workflows: clean context after workflow form action (#39030)
```

#27 - 04 février 2020 11:22 - Serghei Mihai

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

#28 - 06 février 2020 11:10 - Thomas Noël

Fichiers

0001-tests-test-lazy-workflow-data.patch	2,54 ko	27 janvier 2020	Serghei Mihai
form-test-donnees-de-traitement.wcs	1,48 ko	28 janvier 2020	Serghei Mihai
workflow-wf-demande-renseignement-voirie-pour-debug.wcs	16,5 ko	28 janvier 2020	Serghei Mihai
0001-tests-test-lazy-eval-from-multiple-sources-39030.patch	1,59 ko	30 janvier 2020	Serghei Mihai
0001-tests-test-lazy-eval-from-multiple-sources-39030.patch	4,49 ko	02 février 2020	Serghei Mihai
0001-workflows-clean-context-after-workflow-form-action-3.patch	3,86 ko	02 février 2020	Frédéric Péters