

## nettoyer le cache des variables de substitution avant de rentrer dans le workflow

24 avril 2019 12:17 - Nicolas Roche

<b>Statut:</b>	Fermé	<b>Début:</b>	24 avril 2019
<b>Priorité:</b>	Normal	<b>Echéance:</b>	
<b>Assigné à:</b>	Nicolas Roche	<b>% réalisé:</b>	0%
<b>Catégorie:</b>		<b>Temps estimé:</b>	0:00 heure
<b>Version cible:</b>		<b>Planning:</b>	Non
<b>Patch proposed:</b>	Oui		

**Description**

cf #32461 pour le cas réel; simplifié le bug serait identifié via :

```
def test_unfeed_structured_list_when_condition_changes(pub):
    """
    simulate selection of a structured list via condition on from,
    followed by a firstof evaluation on workflow in order to get
    the a structured value from the seleted list.
    """
    # workflow
    Workflow.wipe()
    workflow = Workflow(name='test')
    workflow.backoffice_fields_formdef = WorkflowBackofficeFieldsFormDef(workflow)
    workflow.backoffice_fields_formdef.fields = [
        fields.StringField(
            id='bo1', label='first text', type='string', varname='firstof_text'),
        fields.StringField(
            id='bo2', label='first more', type='string', varname='firstof_more')
    ]

    # status 1 having backoffice fields
    st1 = workflow.add_status('Status1', 'st1')
    setbo = SetBackofficeFieldsWorkflowStatusItem()
    setbo.parent = st1
    setbo.fields = [
        {'field_id': 'bo1',
         'value': '{% firstof form_var_listA form_var_listB %}'},
        {'field_id': 'bo2',
         'value': '{% firstof form_var_listA_more form_var_listB_more %}'}
    ]
    st1.items.append(setbo)
    workflow.store()

    # form providing mainly 2 lists having a condition
    items_A = [dict(id=i, text='A%s' % i, more='moreA%s' % i) for i in range(3)]
    items_B = [dict(id=i, text='B%s' % i, more='moreB%s' % i) for i in range(3)]
    formdef = create_formdef()
    formdef.fields = [
        fields.ItemField(id='1', varname='choice', items=['A', 'B'],
                        type='item', label='list to choice'),
        fields.ItemField(id='2', varname='listA', type='item', label='list A',
                        data_source={'type': 'formula', 'value': str(items_A)},
                        condition={'type': 'python',
                                  'value': 'form_var_choice_raw == "A"'}),
        fields.ItemField(id='3', varname='listB', type='item', label='list B',
                        data_source={'type': 'formula', 'value': str(items_B)},
                        condition={'type': 'python',
                                  'value': 'form_var_choice_raw == "B"'}))
    ]
    formdef.confirmation = False
    formdef.workflow_id = workflow.id
```

```

formdef.store()
formdef.data_class().wipe()

# get form
create_user_and_admin(pub)
resp = get_app(pub).get('/test/')

# user selection on form
resp.form['f1'].value = 'B' # change condition: switch on list B
resp.form['f2'].value = '1' # make 'A1' choice on list A
resp.form['f3'].value = '2' # make 'B2' choice on list B

# submit form
resp = resp.form.submit('submit').follow()
assert 'The form has been recorded' in resp.body

# assert we get a new formdata
assert formdef.data_class().count() == 1
formdata = formdef.data_class().select()[0]
assert formdata.data['1'] == 'B' # choice is list B
assert formdata.data.get('2') is None # list A is hidden
assert formdata.data['3'] == '2' # B2 on list B

# assert firstof value are not computed using previous cached values
assert formdata.data['bo1'] == 'B2' # B2 on list B
assert formdata.data['bo2'] == 'moreB2' # B2 on list B // <--- le bug: ici on a 'moreA1'

```

Le calcul du gabarit ('firstof') basé sur les valeurs supplémentaires des listes structurées utilise des valeurs invalidées lors du changement de la condition d'affichage des listes.

Il s'agit d'un problème avec les valeurs temporaires du système de substitution, et il convient de nettoyer ces valeurs du cache avant de rentrer dans le workflow.

## Révisions associées

### Révision fbc1d9e1 - 02 mai 2019 11:08 - Nicolas Roche

forms: clean form processing context before running workflow actions (#32558)

## Historique

### #2 - 24 avril 2019 13:02 - Nicolas Roche

- *Tracker changé de Support à Development*

### #3 - 24 avril 2019 13:02 - Nicolas Roche

- *Statut changé de Nouveau à Solution proposée*

Voici le patch "Correction barbare" proposé par Fred.

Je constate que le test passe également si on ne dé-provisionne que sur 'ConditionVars'. Svp, donnez-moi votre avis car je me sens un peu dépassé sur ce sujet.

### #4 - 24 avril 2019 13:03 - Nicolas Roche

- *Fichier 0001-unfeed-clean-cached-values-of-substitution-variables.patch ajouté*

- *Patch proposed changé de Non à Oui*

### #5 - 24 avril 2019 14:03 - Nicolas Roche

- *Projet changé de Tours à w.c.s.*

(désolé pour toutes ces fausses manip autour du ticket)

### #6 - 24 avril 2019 14:24 - Thomas Noël

- *Projet changé de w.c.s. à Tours*

Sur la forme, y'a trop de commentaires inutiles dans le test (pour moi tu peux tous les supprimer, le code s'explique lui même à un connaisseur de

wcs).

Sans doute peux-tu éviter le disgracieux (à mon goût) :

```
items_A = [dict(id=i, text='A%s' % i, more='moreA%s' % i) for i in range(3)]
```

et remplacer par un simple

```
items_A = [{'id': '1', 'text': 'A1', 'more': 'moreA1'}]
```

même chose pour B (au passage, pas besoin d'avoir trois choix, c'est pas ce qu'on teste ici).

Sur le fond, comme les "unfeed" ressemble à ce qui est fait dans `submitted_existing` (quand on enregistre un formdata modifié, et non un nouveau formdata comme ici), Frédéric proposait dans #32461 de déplacer du code. Mais je ne vois pas bien où factoriser l'affaire, donc j'aurai tendance à dire que le patch est déjà pas mal ainsi.

Allez, nettoyer un peu le test, puis pousser dans une branche `wip/32558-unfeed-formdata-before-perform-workflow`, et on verra si y'a pas de soucis par ailleurs, déjà.

#### #7 - 24 avril 2019 14:35 - Nicolas Roche

- Fichier `test_on_string_field.py` ajouté

Thomas, je prend en compte tes remarques asap.

Juste pour préciser le contexte : on ne peut pas reproduire avec des champs textes.

Pour moi le bug est lié au listes structurées.

(cf le test joint)

#### #8 - 24 avril 2019 15:57 - Thomas Noël

- *Projet changé de Tours à w.c.s.*

- *Priorité changé de Haut à Normal*

#### #9 - 24 avril 2019 16:03 - Nicolas Roche

- Fichier `0001-unfeed-clean-cached-values-of-substitution-variables.patch` ajouté

- *Projet changé de w.c.s. à Tours*

- *Priorité changé de Normal à Haut*

Voici le patch nettoyé.

#### #10 - 24 avril 2019 16:04 - Nicolas Roche

- *Projet changé de Tours à w.c.s.*

#### #11 - 24 avril 2019 18:22 - Nicolas Roche

- Fichier `test_on_edit.py` ajouté

Voici un second test qui met en évidence qu'effectivement le même bug apparaît lors de l'édition (dans `submitted_existing`).

(ci-joint `test_on_edit.py`)

#### #12 - 24 avril 2019 19:08 - Nicolas Roche

- Fichier `0002-unfeed-clean-cached-values-of-substitution-variables.patch` ajouté

et voici un second patch qui déplace le code pour que l'unfeed s'applique dans les deux situations.

(mais peut-être qu'il vaut mieux ouvrir un second ticket ?)

#### #13 - 26 avril 2019 15:04 - Thomas Noël

Alors, je dirais :

- faire un seul patch des deux
- remplacer le "firstof" dans le premier test, par un "...vs...", car le firstof cache la valeur de la seconde variable (et on aimerait la vérifier)
- la factorisation de "remove\_previous\_formdata\_from\_substitution\_variables" ne me semble pas pertinente, ça obscurcit le code, surtout sur `submitted_existing` où on fait ensuite un "# and add new one", on comprend plus rien
- sur le message de commit, plutôt "clean live substitutions vars before executing workflow" (l'idée étant que les "live substitutions vars" sont celles qu'on utilise lorsqu'on joue dans les calculs de conditions & co, et on ne veut plus les avoir quand le formdata est ok)

#### #14 - 26 avril 2019 17:33 - Nicolas Roche

- Fichier 0001-unfeed-clean-live-substitutions-vars-before-executin.patch ajouté

j'ai appliqué toutes les corrections énoncées ci-dessus.

#### #15 - 30 avril 2019 12:37 - Nicolas Roche

- *Priorité changé de Haut à Normal*

#32461 a une solution de contournement (sur l'instance de test) utilisant if/elif/endif à la place de firstof.

#### #16 - 30 avril 2019 23:33 - Thomas Noël

- *Statut changé de Solution proposée à Solution validée*

Le code me semble bon.

Mais j'aime bien pinailler.

Pour le message de commit, on commence par "xxxx:" où xxxx est le module ou le truc sur lequel on va agir. Et ensuite une phrase qui explique un peu. Ici on peut s'inspirer de [#30942](#) et je propose :

```
forms: don't let old context vars get used by workflow just after store formdata or edit action (#32558)
```

Ça dépasse un peu mais je sais pas faire plus court. (tu noteras certainement que je parle anglais like a spanish cow)

Ensuite il y a le nom de tes tests : (je suis *vraiment* pénible, tu en fait ce que tu veux de cette remarque)

- test\_unfeed\_structured\_list\_when\_condition\_changes : ça passe mais on sent que le test est écrit suite à la correction d'un bogue :) c'est pas toujours ce qu'on veut exprimer, on voudrait plutôt dire "voilà je teste ça". Et ici, je dirais :  
test\_backoffice\_fields\_just\_after\_conditional\_form\_submit
- test\_form\_edit\_and\_backoffice\_field\_condition\_change même chose, plutôt test\_backoffice\_fields\_just\_after\_conditional\_form\_edit\_action

Enfin le commentaire que tu laisses dans wcs/forms/root.py n'est plus tellement bon : « # remove previous formdata from substitution variables » doit devenir un truc comme « # remove previous vars and formdata from substitution variables »

Voilà... tu vois je pinaille :) Il faut surtout modifier le message de commit, et ensuite, tu pousses. On tagguera le plus tôt possible pour que ça vive en recette une semaine au moins.

#### #17 - 01 mai 2019 16:21 - Frédéric Péters

```
forms: don't let old context vars get used by workflow just after store formdata or edit action (#32558)
```

```
forms: clean form processing context before running workflow actions (#32558)
```

#### #18 - 02 mai 2019 11:12 - Nicolas Roche

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

Pas de soucis Thomas, toutes tes pinailleries me sont profitables.

```
commit fbcd9e1ea6a96af76d36037e769caf8a9808d26
Author: Nicolas ROCHE <nroche@entrouvert.com>
Date: Wed Apr 24 12:18:54 2019 +0200
```

```
forms: clean form processing context before running workflow actions (#32558)
```

#### #19 - 02 mai 2019 16:15 - Frédéric Péters

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

#### #20 - 17 juillet 2019 17:56 - Brice Mallet

- *Statut changé de Solution déployée à Fermé*

### Fichiers

0001-unfeed-clean-cached-values-of-substitution-variables.patch	4,6 ko	24 avril 2019	Nicolas Roche
test_on_string_field.py	2,53 ko	24 avril 2019	Nicolas Roche

0001-unfeed-clean-cached-values-of-substitution-variables.patch	4,14 ko	24 avril 2019	Nicolas Roche
test_on_edit.py	3,44 ko	24 avril 2019	Nicolas Roche
0002-unfeed-clean-cached-values-of-substitution-variables.patch	6,19 ko	24 avril 2019	Nicolas Roche
0001-unfeed-clean-live-substitutions-vars-before-executin.patch	8,21 ko	26 avril 2019	Nicolas Roche