w.c.s. - Development #58207

autosave peut sauver des données invalides

26 octobre 2021 17:15 - Benjamin Dauvergne

Statut:	Rejeté	**Début:**	26 octobre 2021
Priorité:	Normal	**Echéance:**	
Assigné à:	Benjamin Dauvergne	**% réalisé:**	0%
Catégorie:		**Temps estimé:**	0:00 heure
Version cible:			
Patch proposed:	Oui	**Planning:**	Non

Description

Dans autosave, on a le code suivant :

```
        form = self.create_form(page=page)
        data = self.formdef.get_data(form)
```

or get_data ne vérifie pas si le widget est en état d'erreur (ex. un ItemField dans la source de donnée est en erreur pendant un instant, la valeur soumise n'est plus valide et widget.parse() renvoie None) avant d'utiliser la valeur renvoyée par widget.parse(...), si jamais l'autosave se passe après une soumission il est possible d'écraser des valeurs valides par des valeurs invalides.

Historique

#1 - 26 octobre 2021 17:15 - Benjamin Dauvergne

- Sujet changé de autosave se peut sauver des données invalides à autosave peut sauver des données invalides

#3 - 26 octobre 2021 18:20 - Benjamin Dauvergne

- Assigné à mis à Benjamin Dauvergne

#4 - 26 octobre 2021 18:24 - Benjamin Dauvergne

Première idée de correction :

```
diff --git a/wcs/formdef.py b/wcs/formdef.py
index 2997c836..d2db6202 100644
--- a/wcs/formdef.py
+++ b/wcs/formdef.py
@@ -818,11 +818,11 @@ class FormDef(StorableObject):

     def get_data(self, form):
         d = {}
         for field in self.fields:
             widget = form.get_widget('f%s' % field.id)
-            if widget:
+            if widget and not widget.has_error():
                 d.update(self.get_field_data(field, widget))
         return d

     def export_to_json(self, include_id=False, indent=None, anonymise=True):
         charset = get_publisher().site_charset
```

on va voir ce que ça donne au niveau des tests, j'ai quand même réussi à simuler un autosave un peu sauvage lié à une datasource (j'aurai pu utiliser n'importe quoi comme un StringField avec validation).

#5 - 26 octobre 2021 18:35 - Frédéric Péters

Je ne sais pas ce que diront les tests mais c'est prévu que des données invalides puissent être enregistrées, pour que l'usager puisse revenir sur une page dans l'état où il était occupé à la saisir, la validation s'opérant en sortie de page.

#6 - 26 octobre 2021 19:45 - Frédéric Péters

Les tests passant je viens de pousser un test explicitant le fonctionnement attendu de sauvegarde de données avant leur validation. (test_form_autosave_with_invalid_data).

#7 - 26 octobre 2021 19:52 - Frédéric Péters

i.e. je pense qu'il faut être plus fin, attraper la situation précise ("échec de la source de données"), ça pourrait être une exception remontée jusqu'à l'autosave qui dans ce cas s'interromperait (= je ne pense pas qu'il y ait à tenter une sauvegarde partielle "tous les champs sauf ceux où la source de données a échoué").

#8 - 26 octobre 2021 20:57 - Benjamin Dauvergne

Frédéric Péters a écrit :

> i.e. je pense qu'il faut être plus fin, attraper la situation précise ("échec de la source de données"), ça pourrait être une exception remontée jusqu'à l'autosave qui dans ce cas s'interromperait (= je ne pense pas qu'il y ait à tenter une sauvegarde partielle "tous les champs sauf ceux où la source de données a échoué").

Ok, je vais limiter ça à ItemField et ItemsField, par contre je ne pense pas qu'on ait besoin de remonter jusqu'à la détection de l'exception car on a que deux cas d'invalidité :

- le champ est requis et on soumet la valeur vide, on peut sauver ça,
- ou soumet une valeur non vide mais elle ne valide pas, ça veut forcément dire que la valeur soumise ne correspond pas à une des valeurs dans widget.options, dans tous les cas on ne veut pas de cette valeur à ce moment là, une valeur invalide ne devrait pas être possible dans un Item/ItemsField de toute façon (contrairement aux autres champs en saisie plus ou moins libre) même de façon temporaire.

#9 - 26 octobre 2021 20:58 - Benjamin Dauvergne

- Fichier 0001-formdef-do-not-load-data-from-item-s-widgets-with-er.patch ajouté

- Tracker changé de Bug à Development

- Statut changé de Nouveau à Solution proposée

- Patch proposed changé de Non à Oui

Rebasé sur main et donc avec la restriction aux champs ItemField et ItemsField quand la valeur est non nulle/vide mais qu'elle ne valide pas.

#10 - 26 octobre 2021 22:04 - Benjamin Dauvergne

- Statut changé de Solution proposée à En cours

#11 - 26 octobre 2021 22:37 - Benjamin Dauvergne

- Fichier 0001-formdef-do-not-load-data-from-item-s-widgets-with-er.patch ajouté

- Statut changé de En cours à Solution proposée

Sans dépendre de Widget.parse(), on ne peut pas l'appeler deux fois.

#13 - 12 novembre 2021 14:55 - Frédéric Péters

J'ai l'impression qu'on peut ignorer tout ça à partir du moment où

> si jamais l'autosave se passe après une soumission

cette situation ne peut plus arriver que c'est l'objet de #58276 ?

Aussi, que ça serait de toute façon à revoir après #56824 ?

#14 - 22 novembre 2021 10:42 - Benjamin Dauvergne

- Statut changé de Solution proposée à En cours

#15 - 23 novembre 2021 11:30 - Benjamin Dauvergne

Frédéric Péters a écrit :

> cette situation ne peut plus arriver que c'est l'objet de #58276 ?

Le ticket a pour titre "limiter au maximum" ce qui est un grand progrès par rapport à "on s'en fout complètement", donc non je ne pourrai pas jurer que ça ne peut plus arriver; la remarque suivante est plus intéressante.

> Aussi, que ça serait de toute façon à revoir après #56824 ?

Ça devient effectivement inutile après #56824 qui est allé plus loin que prévu initialement (ici ça fera foirer l'autosave complètement avec une erreur

"form deserialization failed"), donc oui, si on peut avoir #56824 vite on peut rejeter celui-ci.

#16 - 05 janvier 2022 17:56 - Benjamin Dauvergne

- Statut changé de En cours à Rejeté

Je me rejette dans le mazout; les autres tickets liés à #58260 devraient corriger ce problème.

Fichiers

0001-formdef-do-not-load-data-from-item-s-widgets-with-er.patch	4,52 ko	26 octobre 2021	Benjamin Dauvergne
0001-formdef-do-not-load-data-from-item-s-widgets-with-er.patch	4,41 ko	26 octobre 2021	Benjamin Dauvergne