

## w.c.s. - Development #39356

### faire en sorte que FormData.get\_url retourne la bonne URL pour les brouillons

28 janvier 2020 18:21 - Benjamin Dauvergne

|   |                 |                      |   |
|---|-----------------|----------------------|---|
| <b>Statut:</b>  | Fermé           | <b>Début:</b>        | 28 janvier 2020                           |
| <b>Priorité:</b>  | Normal          | <b>Echéance:</b>     |   |
| <b>Assigné à:</b>   | Frédéric Péters | <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             |                      |   |
| <b>Description</b>  |                 |                      |   |
| Pour un draft l'URL ne doit pas avoir de slash à la fin, et pour un draft en soumission backoffice l'URL de base n'est pas la même.   |                 |                      |   |
| Soulevé lors du développement de l'action initialisation d'un brouillon, il est bien pratique dans ce cas de pouvoir rediriger vers celui-ci sans trop réfléchir (via form_links_{varname}.form_url). |                 |                      |   |
| <b>Demandes liées:</b>  |                 |                      |   |
| Bloque w.c.s. - Development #33186: Initialisation d'un brouillon   |                 | <b>Fermé</b>         | <b>17 mai 2019</b> <b>17 janvier 2020</b> |

### Révisions associées

#### Révision d49de08b - 01 février 2020 15:00 - Frédéric Péters

misc: load drafts from the form URL, with trailing slash (#39356)

### Historique

#### #1 - 28 janvier 2020 19:32 - Benjamin Dauvergne

J'ai deux pistes, modifier FormData.get\_url() (et ça va changer d'autres trucs) ou modifier FormStatusPage.\_q\_index() pour dans le cas d'un draft renvoyer vers self.filled.get\_url().rstrip('/') (c'est ce qui est fait dans FormBackOfficeStatusPage.\_q\_index après une vérification des rôles de l'utilisateur par rapport aux rôles de soumission backoffice); une préférence ?

#### #2 - 29 janvier 2020 00:38 - Benjamin Dauvergne

Bon en fait le code existe déjà dans FormPage.\_q\_lookup() c'est juste qu'il ne marche pas car \_q\_lookup() n'est pas autorisé par quixote à retourner directement une réponse.

```
class FormPage:
    ...
    def _q_lookup(self, component):
        import pdb
        pdb.set_trace()
        try:
            filled = self.formdef.data_class().get(component)
        except KeyError:
            raise errors.TraversalError()

        if not filled.is_draft():
            if get_request().is_in_backoffice():
                get_session().message = ('error', _('This form has already been submitted.'))
                return redirect(get_publisher().get_backoffice_url() + '/submission/')
            return PublicFormStatusPage(self.formdef, filled)

        # restore draft
        session = get_session()
        if not (get_request().is_in_backoffice() and filled.backoffice_submission):
            if session.is_anonymous_submitter(filled):
                pass
            elif session.user:
                if str(session.user) != str(filled.user_id):
                    raise errors.AccessUnauthorizedError()
            else:
                raise errors.AccessUnauthorizedError()

        if get_request().get_query() == 'remove-draft':
            filled.remove_self()
            return redirect(get_publisher().get_root_url())
```

```

magictoken = randbytes(8)
filled.feed_session()
form_data = filled.data
for field in filled.formdef.fields:
    if not field.id in form_data:
        continue
    if form_data[field.id] is None:
        # remove keys that were not set, this is required when we restore a
        # draft from SQL (where all columns are always defined).
        del form_data[field.id]
        continue
    if field.type == 'file':
        # add back file to session
        tempfile = session.add_tempfile(form_data[field.id])
        form_data[field.id].token = tempfile['token']
form_data['is_recalled_draft'] = True
form_data['draft_formdata_id'] = filled.id
form_data['page_no'] = filled.page_no
session.add_magictoken(magictoken, form_data)

if get_request().is_in_backoffice():
    return redirect('./?mt=%s' % magictoken)
else:
    return redirect('%s?mt=%s' % (filled.formdef.get_url(), magictoken))

```

C'est appelé par `Directory._q_traverse` :

```

def _q_traverse(self, path):
    """(path: [string]) -> object

    Traverse a path and return the result.
    """
    assert len(path) > 0
    component = path[0]
    path = path[1:]
    name = self._q_translate(component)
    if name is not None:
        obj = getattr(self, name)
    else:
        obj = self._q_lookup(component)
    if obj is None:
        raise TraversalError(private_msg=('directory %r has no component '
                                         '%r' % (self, component)))

    if path:
        if hasattr(obj, '_q_traverse'):
            return obj._q_traverse(path)
        else:
            raise TraversalError <-- ça tombe systématiquement en 404 car _q_
    elif hasattr(obj, '__call__'):
        return obj()
    else:
        return obj

```

Tout ce code devrait être déplacé dans `PublicFormStatusPage._q_index()`.

### #3 - 29 janvier 2020 06:44 - Frédéric Péters

- Assigné à mis à Frédéric Péters

Je vais regarder ça.

### #4 - 29 janvier 2020 08:32 - Benjamin Dauvergne

- Assigné à Frédéric Péters supprimé

J'ai une solution à peu près minimale il me semble, ça permet de respect la signature attendu par `_q_traverse()` quand il appelle `_q_lookup()` le retour doit posséder une méthode `_q_traverse()` si `path` n'est pas vide (ce qui est le cas quand l'URL se termine par un slash).

```

diff --git wcs/forms/root.py wcs/forms/root.py
index 960a65cb..b6d0e30a 100644
--- wcs/forms/root.py
+++ wcs/forms/root.py

```

```

@@ -1329,10 +1329,20 @@ class FormPage(Directory, FormTemplateMixin):
    form_data['page_no'] = filled.page_no
    session.add_magictoken(magictoken, form_data)

+     class Redirector:
+         def __init__(self, url):
+             self.url = url
+
+         def _q_traverse(self, path):
+             return self()
+
+         def __call__(self):
+             return redirect(self.url)
+
    if get_request().is_in_backoffice():
-         return redirect('./?mt=%s' % magictoken)
+         return Redirector('./?mt=%s' % magictoken)
    else:
-         return redirect('%s?mt=%s' % (filled.formdef.get_url(), magictoken))
+         return Redirector('%s?mt=%s' % (filled.formdef.get_url(), magictoken))

class RootDirectory(AccessControlled, Directory):

```

#### #5 - 29 janvier 2020 09:12 - Frédéric Péters

- Assigné à mis à Frédéric Péters

Je préfère regarder.

#### #6 - 30 janvier 2020 12:17 - Frédéric Péters

- Bloque Development #33186: Initialisation d'un brouillon ajouté

#### #7 - 01 février 2020 14:41 - Frédéric Péters

- Fichier 0001-misc-load-drafts-from-the-form-URL-with-trailing-sla.patch ajouté

- Statut changé de Nouveau à Solution proposée

- Patch proposed changé de Non à Oui

Passage de fond pour ne plus avoir ce truc d'URL sans / final.

#### #8 - 01 février 2020 22:30 - Benjamin Dauvergne

- Statut changé de Solution proposée à Solution validée

#### #9 - 02 février 2020 19:23 - Frédéric Péters

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

```

commit d49de08b18343ecb53310ad0a905ee9b589b9fe5
Author: Frédéric Péters <fpeters@entrouvert.com>
Date: Sat Feb 1 14:35:04 2020 +0100

```

```

misc: load drafts from the form URL, with trailing slash (#39356)

```

#### #10 - 04 février 2020 11:46 - Frédéric Péters

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

### Fichiers

---

|   |         |                 |                 |
|---|---------|-----------------|-----------------|
| 0001-misc-load-drafts-from-the-form-URL-with-trailing-sla.patch | 28,8 ko | 01 février 2020 | Frédéric Péters |
|---|---------|-----------------|-----------------|