

## w.c.s. - Bug #9839

### Activer le ckeditor pour le contenu des champs de type "commentaire"

02 février 2016 15:24 - Frédéric Péters

<b>Statut:</b>	Fermé	<b>Début:</b>	02 février 2016
<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>	v1.33	<b>Planning:</b>	
<b>Patch proposed:</b>	Oui		
<b>Description</b>			

#### Révisions associées

##### Révision 7c071d52 - 16 février 2016 11:14 - Frédéric Péters

misc: use wysiwyg widgets for comments field (#9839)

#### Historique

##### #1 - 02 février 2016 15:25 - Frédéric Péters

Rapidement fait pendant la réunion hier mais à vérifier, compléter de tests, etc.

```
--- a/wcs/fields.py
+++ b/wcs/fields.py
@@ -494,8 +494,12 @@ class CommentField(Field):
     pass

     def fill_admin_form(self, form):
-        form.add(TextWidget, 'label', title=_('Label'), value = self.label,
-                  required = True, cols = 70, rows = 3, render_br = False)
+        if self.label and not self.label.startswith('<') and '\n\n' in self.label:
+            form.add(TextWidget, 'label', title=_('Label'), value=self.label,
+                      required=True, cols=70, rows=3, render_br=False)
+        else:
+            form.add(WysiwygTextWidget, 'label', title=_('Label'), value=self.label,
+                      required=True)
+        form.add(StringWidget, 'extra_css_class', title = _('Extra classes for CSS styling'),
+                  value=self.extra_css_class, size=30, advanced=(not self.extra_css_class))
```

##### #2 - 12 février 2016 09:11 - Frédéric Péters

- Fichier 0001-misc-use-wysiwyg-widgets-for-comments-field-9839.patch ajouté
- Statut changé de Nouveau à En cours
- Patch proposed changé de Non à Oui

##### #3 - 12 février 2016 11:25 - Thomas Noël

Si tout le monde est bien conscient que cela risque de faire "exploser" les commentaires qui avait été rédigés avec du HTML (ckeditor ayant fortement tendance à nettoyer le HTML à sa façon), je suis ok.

Je serais d'avis de faire un tour des champs commentaires sur notre prod, quand même.

##### #4 - 12 février 2016 11:35 - Frédéric Péters

Ça ne va rien faire exploser de manière spontanée. À l'édition d'un champ commentaire dont l'HTML serait défectueux, celui-ci sera d'abord remis d'aplomb et ça correspondra au rendu DOM qui était de toute façon déterminé par le navigateur.

exemple : `<p>comm<p>hello<li>TOTO donnera <p>comm</p><p>hello</p><ul><li>TOTO</li></ul>`.

Ce qui ne sera plus possible, par contre, c'est vrai, c'est commencer de l'HTML dans un premier champ commentaire et le terminer dans un second; genre un premier je mets `<div style="background: red;">` et un autre je mets `</div>`. Si on veut pouvoir continuer à faire ça, on doit modifier la détection sur l'usage du ckeditor pour l'activer uniquement si les balises sont correctement balancées.

On a le temps de discuter, je voudrais de toute façon d'abord avoir [#9837](#).

**#5 - 12 février 2016 11:51 - Thomas Noël**

(note quand même : je pense aussi aux commentaires avec des [if-any ...] qui risquent de devenir un peu moins clairs, faudra quand même bien tester tout ça)

**#6 - 12 février 2016 12:04 - Frédéric Péters**

On peut aussi ne pas activer le ckeditor si on voit un [.

**#7 - 12 février 2016 12:05 - Frédéric Péters**

Frédéric Péters a écrit :

On peut aussi ne pas activer le ckeditor si on voit un [.

Ce qui est sans doute excessif, mettons plutôt, si on voit un [if

**#8 - 12 février 2016 13:10 - Thomas Noël**

pendant qu'on y est : [is, [if et [for ... ou bien plus bêtement [end] ?

**#9 - 12 février 2016 15:17 - Thomas Noël**

- Version cible mis à v1.33

**#10 - 12 février 2016 16:58 - Frédéric Péters**

- Fichier 0001-misc-use-wysiwyg-widgets-for-comments-field-9839.patch ajouté

Oui, [end].

**#11 - 12 février 2016 17:03 - Frédéric Péters**

Je viens de faire l'inventaire de la prod, il n'y a pas d'utilisation d'html divisé sur deux champs. (ma méthode : regarder ce qui commence par un < mais ne finit pas par un >, les résultats sont du genre <strong>Important : </strong>bla bla bla.

**#12 - 15 février 2016 10:25 - Thomas Noël**

Du coup je ne sais plus quoi penser ;)

**#13 - 16 février 2016 11:10 - Thomas Noël**

Ack

**#14 - 16 février 2016 11:14 - Frédéric Péters**

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

```
commit 7c071d5248a512ce14fd668fd09dc203d7c0be1c
Author: Frédéric Péters <fpeters@entrouvert.com>
Date: Thu Feb 4 17:52:14 2016 +0100
```

```
misc: use wysiwyg widgets for comments field (#9839)
```

**#15 - 19 février 2016 00:40 - Thomas Noël**

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

**#16 - 24 février 2016 17:12 - Thomas Noël**

Je crois qu'on s'est trompé, le wysiwyg reste quand il y a un [end]

Test qui ne passe pas:

```
diff --git a/tests/test_admin_pages.py b/tests/test_admin_pages.py
index 0847a37..8ca7592 100644
--- a/tests/test_admin_pages.py
+++ b/tests/test_admin_pages.py
@@ -986,6 +986,13 @@ def test_form_edit_comment_field(pub):
     resp = app.get('/backoffice/forms/1/fields/1/')
     assert 'WysiwygTextWidget' in resp.body

+ # starting with a < but with ezt
+ formdef.fields = [fields.CommentField(id='1', type='comment',
```

```
+     label='<p>a comment [if-any test]with ezt[end]</p>']
+     formdef.store()
+     resp = app.get('/backoffice/forms/1/fields/1/')
+     assert 'WysiwygTextWidget' not in resp.body
+
def test_form_legacy_int_id(pub):
    create_superuser(pub)
    create_role()
```

## Fichiers

---

0001-misc-use-wysiwyg-widgets-for-comments-field-9839.patch	2,71 ko	12 février 2016	Frédéric Péters
0001-misc-use-wysiwyg-widgets-for-comments-field-9839.patch	2,76 ko	12 février 2016	Frédéric Péters