

w.c.s. - Development #19442

passer en template django dès présence de {{ ou {%

14 octobre 2017 18:23 - Thomas Noël

Statut:	Fermé	Début:	14 octobre 2017
Priorité:	Normal	Echéance:	
Assigné à:		% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	
Patch proposed:	Oui		
Description <ul style="list-style-type: none">• dans les mails• dans les champs "compute"• les "variadic_url"			

Révisions associées

Révision f626c45c - 01 décembre 2017 16:38 - Thomas Noël

add new Template system, authorizing Django and ezt syntaxes (#19442)

Révision 7d92cf29 - 01 décembre 2017 16:38 - Thomas Noël

use Django/ezt templates on computed expressions (#19442)

Révision 18c9258e - 01 décembre 2017 16:38 - Thomas Noël

use Django/ezt templates in messages (email, sms, journal) (#19442)

Révision 3906f827 - 01 décembre 2017 17:10 - Thomas Noël

misc: use Django/ezt templates in variadic URLs (#19442)

Révision 78090604 - 01 décembre 2017 17:11 - Thomas Noël

workflows: allow Django formatted receipient in sendmail action (#19442)

Révision e693c220 - 01 décembre 2017 17:11 - Thomas Noël

tests: add some Django/ezt templates tests (#19442)

Historique

#1 - 15 octobre 2017 01:09 - Thomas Noël

- Description mis à jour

#2 - 15 octobre 2017 03:01 - Thomas Noël

- Fichier 0001-add-new-Template-system-authorizing-Django-and-ezt-s.patch ajouté

- Fichier 0002-use-Django-ezt-templates-on-computed-expressions-194.patch ajouté

- Patch proposed changé de Non à Oui

wip, le début, pour valider ma piste :

- 0001 : ajoute un nouvel objet Template qui mange du Django et de l'ezt et effectue la vérification syntaxique et le rendu
- 0002 : utilisation par tout ce qui est "compute", avec par ricochet prise en charge dans le widget

Dans 0001, j'ai posé Template dans qommon/template.py mais ça pourrait être ailleurs, j'avais juste pas trop d'inspiration à cet instant.

Il y a dans 0002 des ajouts dans les tests : là où on vérifiait la partie ezt, on ajoute des vérifications identiques en Django. Je n'ai pas déplacé le "compute" qui reste dans WorkflowStatusItem, pour éviter de rendre le patch trop intrusif et moins lisible. Ça pourrait être fait ensuite.

#3 - 16 octobre 2017 10:03 - Frédéric Péters

Je n'ai pas lu attentivement mais ça m'a l'air ok.

#4 - 18 octobre 2017 00:18 - Thomas Noël

- Fichier 0003-use-Django-ezt-templates-in-messages-email-sms-journ.patch ajouté
- Fichier 0002-use-Django-ezt-templates-on-computed-expressions-194.patch ajouté
- Fichier 0001-add-new-Template-system-authorizing-Django-and-ezt-s.patch ajouté

Étape suivante !

0001: aucune modif.

0002: j'avais oublié la gestion des "compute" dans le champ de signature des webservices, c'est ajouté. Avec un str() pour récupérer le résultat car sign_url n'aime pas l'unicode renvoyé par le render Django.

0003: utilisation dans les messages, mails, sms, commentaires dans le journal, etc.

Je pensais faire un truc un peu plus serré, mais modifier template_on_formdata a fait remonter pas mal de dépendances, que j'ai donc adaptées petit à petit. Mais ce patch 0003 reste encore lisible et cohérent à mon goût. Ca remplace clairement les appels ezt par l'utilisation de qommon.template.Template

Il y a de l'historique pas facile à gérer au niveau par exemple des JournalEvolutionPart, avec une détection du mode HTML si ça commence par "<", un encodage des variables dans ce cas, etc... Je n'ai pas réussi à m'en défaire, et donc si on doit utiliser du Django ici il faut jouer avec des {{truc|safe}}.

Plus dommage, et là encore j'ai pas de solution, c'est la même chose pour les mails. Comme les mails sont normalement du texte pur, on pourrait dire à Django de faire du "safe" systématique... mais d'abord ça n'existe pas dans Django (sauf à ajouter du "autoescape on" dans le template) et y'a un mic mac avec le ezt.FORMAT_RAW/FORMAT_URL et le processeur de contexte, que je n'ai pas réussi à démêler.

Malgré tout, on le voit dans les tests, ça marche bien. Sans doute faudrait-il poser ça sur une machine de dev que des bonnes volontés viendrait tester (rêvons).

(me reste à faire la partie 0004 variadic)

#5 - 18 novembre 2017 20:50 - Thomas Noël

- Fichier 0001-add-new-Template-system-authorizing-Django-and-ezt-s.patch ajouté
- Fichier 0002-use-Django-ezt-templates-on-computed-expressions-194.patch ajouté
- Fichier 0003-use-Django-ezt-templates-in-messages-email-sms-journ.patch ajouté

Rebasés sans dommage

#6 - 24 novembre 2017 15:13 - Frédéric Péters

```
def test_template(): #pub):
```

score à retirer.

```
    message = _('fail to render ezt template: %s')
```

s/fail/failure/

```
def null_render(self, context={}):
    return self.value
```

str(self.value) ?

```
- def custom_ezt_email(key, mail_body_data, email_rcpt, **kwargs):
```

Attention custom_ezt_email est utilisé dans le module auquotidien.

```
-     assert isinstance(i, int)
+     if not isinstance(i, int):
+         raise TypeError
```

ok mais rien à faire ici.

```
-     if not '[' in template:
+     if '{{' not in template and '%' not in template and '[' not in template:
+         # no tags, no variables: don't even process formdata
```

une fois on fait ça (template_on_formdata) mais plus loin (template_on_context), pas.

#7 - 27 novembre 2017 16:58 - Thomas Noël

- Fichier 0003-use-Django-ezt-templates-in-messages-email-sms-journ.patch ajouté
- Fichier 0002-use-Django-ezt-templates-on-computed-expressions-194.patch ajouté
- Fichier 0001-add-new-Template-system-authorizing-Django-and-ezt-s.patch ajouté
- Fichier 0001-use-new-custom_template_email-system-19422.patch ajouté

Frédéric Péters a écrit :

```
scorie à retirer.  
s/fail/failure/  
str(self.value) ?
```

ok

Attention custom_ezt_email est utilisé dans le module auquotidien.

pour ça, j'ajoute, poussé en même temps, 0001-use-new-custom_template_email-system-19422.patch sur auquotidien

ok mais rien à faire ici.

En fait si, parce que Django a besoin d'une TypeError (et pas d'une AssertionError) pour comprendre quoi faire avec {{attachments.testfile.url}}, c-a-d aller chercher un attribut quand le getitem ne passe pas (sans ce raise, les tests plantent sur test_register_comment_attachment)

une fois on fait ça (template_on_formdata) mais plus loin (template_on_context), pas.

Yep, petite optimisation ajoutée aussi.

#8 - 27 novembre 2017 22:30 - Thomas Noël

- Fichier 0006-tests-add-some-Django-ezt-templates-tests-19442.patch ajouté
- Fichier 0005-workflows-allow-Django-formatted-receipient-in-sendma.patch ajouté
- Fichier 0004-misc-use-Django-ezt-templates-in-variadic-URLs-19442.patch ajouté

Suite (et fin ? du moins pour la plomberie) dans le 0004 : on accepte la syntaxe Django pour les URLs. Cependant on ne gère pas, en Django, d'aspect magique comme avec ezt : le langage Django est suffisamment complet pour permettre à l'utilisateur s'indiquer s'il veut encoder, ajouter ou pas un /, etc.

Suivent deux plus petits patches :

- 0005 : import/export des destinataires de mail exprimés en django
- 0006 : Un dernier patch aussi pour ajouter des tests à des endroits où il n'y avait que les tests en ezt.

#9 - 28 novembre 2017 13:08 - Benjamin Dauvergne

Thomas Noël a écrit :

En fait si, parce que Django a besoin d'une TypeError (et pas d'une AssertionError) pour comprendre quoi faire avec {{attachments.testfile.url}}, c-a-d aller chercher un attribut quand le getitem ne passe pas (sans ce raise, les tests plantent sur test_register_comment_attachment)

On l'obtiendra pas bêtement de attachment = self.get_attachments()[i] ?

#10 - 28 novembre 2017 16:08 - Thomas Noël

Benjamin Dauvergne a écrit :

Thomas Noël a écrit :

```
En fait si, parce que Django a besoin d'une TypeError (et pas d'une AssertionError) pour comprendre quoi faire avec  
{{attachments.testfile.url}}, c-a-d aller chercher un attribut quand le getitem ne passe pas (sans ce raise, les tests plantent sur  
test_register_comment_attachment)
```

On l'obtiendra pas bêtement de attachment = self.get_attachments()[i] ?

De l'intérêt des relectures, effectivement ! Merci Benj.

C'est donc maintenant juste :

```
def __getitem__(self, i):
-     assert isinstance(i, int)
        attachment = self.get_attachments()[i]
        return AttachmentSubstitutionProxy(self.formdata, attachment)
```

J'ai poussé les patches dans <http://git.entrouvert.org/wcs.git?h=wip/19442-django-template> pour permettre une lecture plus "facile".

#11 - 28 novembre 2017 23:19 - Frédéric Péters

(c'est une relecture basée sur les patches anciens, ce matin dans le train)

Il y a quantité de%:

```
'{' not in string and '%' not in string and '[' not in string
```

ex:

```
wcs/data_sources.py:         if '{{' in url or '%' in url or '[' in url:
wcs/qommon/form.py:         if '{{' in self.url or '%' in self.url or '[' in self.url:
wcs/qommon/form.py:         if self.url and ('{{' in self.url or '%' in self.url or '[' in self.url):
wcs/qommon/misc.py:         if '{{' not in url and '%' not in url and '[' not in url:
wcs/qommon/misc.py:         if '{{' in url or '%' in url:
wcs/workflows.py:         if '{{' not in template and '%' not in template and '[' not in template:
wcs/workflows.py:         if '{{' not in template and '%' not in template and '[' not in template:
wcs/wscalls.py:         if '{{' in url or '%' in url or '[' in url:
```

Possibilité de réunir ça, genre

```
class Template(object):
    @classmethod
    def is_template_string(cls, string):
        return string and ('{{' in string or '%' in string or '[' in string)
```

#12 - 30 novembre 2017 16:25 - Thomas Noël

Frédéric Péters a écrit :

Il y a quantité de :

```
'{' not in string and '%' not in string and '[' not in string
```

Yep. J'ai construit le `Template.is_template_string` et l'ai posé partout où nécessaire. C'est pushforcé sur <http://git.entrouvert.org/wcs.git?h=wip/19442-django-template>

#13 - 01 décembre 2017 16:23 - Frédéric Péters

- Statut changé de Nouveau à En cours

```
+     except TemplateSyntaxError, VariableDoesNotExist:
```

Manquent des parenthèses, et peut-être du coup un test pour vérifier que `VariableDoesNotExist` est bien attrapé; mais même sans le test supplémentaire, ok pour moi que ça soit poussé après l'ajout des parenthèses.

#14 - 01 décembre 2017 17:14 - Thomas Noël

Parenthèses ajoutées, et ces tests (où l'on voit qu'avec ezt ça crashe violemment et pas avec Django, et c'est ainsi) :

```
def test_url_bad_syntax():
    with pytest.raises(EZTException):
        get_variadic_url('[if-any form_avr]https://example.net/[foo]/', {'foo': 'bar'})

    # Django TemplateSyntaxError
    assert get_variadic_url('{% if %}https://www/{ { foo } }',
        {'bar': 'nofoo'}) == '{% if %}https://www/{ { foo } }'

    # Django VariableDoesNotExist
    assert get_variadic_url('{{ foo|default:notexist }}', {}) == '{{ foo|default:notexist }}'
```

#15 - 01 décembre 2017 17:44 - Thomas Noël

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

```
commit e693c2200e56372e3a75f7e714b3819c1fc177ee (HEAD -> master, origin/wip/19442-django-template, origin/mast
```

er, origin/HEAD, wip/19442-django-template)
Author: Thomas NOEL <tnoel@entrouvert.com>
Date: Mon Nov 27 22:26:38 2017 +0100

tests: add some Django/ezt templates tests (#19442)

commit 7809060474a74e8d72479fbed634dfd191788931
Author: Thomas NOEL <tnoel@entrouvert.com>
Date: Mon Nov 27 22:20:11 2017 +0100

workflows: allow Django formatted receipient in sendmail action (#19442)

commit 3906f827cdcb4fd9abb31e173496c51123f384e9
Author: Thomas NOEL <tnoel@entrouvert.com>
Date: Mon Nov 27 22:05:14 2017 +0100

misc: use Django/ezt templates in variadic URLs (#19442)

commit 18c9258e9faf8793bec49d1e15079164119fbbb7
Author: Thomas NOEL <tnoel@entrouvert.com>
Date: Wed Oct 18 00:02:29 2017 +0200

use Django/ezt templates in messages (email, sms, journal) (#19442)

commit 7d92cf296e109c3ce92f255502f02f5190ecd844
Author: Thomas NOEL <tnoel@entrouvert.com>
Date: Sun Oct 15 02:48:32 2017 +0200

use Django/ezt templates on computed expressions (#19442)

commit f626c45cbb51a7abf7bed66a36322ee5ca5ac184
Author: Thomas NOEL <tnoel@entrouvert.com>
Date: Sat Oct 14 18:26:39 2017 +0200

add new Template system, authorizing Django and ezt syntaxes (#19442)

et sur auquotidien

commit fb7f1f78416772a1b8b16f7e89f13119e68f4553 (HEAD -> master, origin/master, origin/HEAD)
Author: Thomas NOEL <tnoel@entrouvert.com>
Date: Mon Nov 27 16:38:57 2017 +0100

use new w.c.s. custom_template_email system (#19422)

#16 - 23 décembre 2018 14:45 - Frédéric Péters

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

Fichiers

0001-add-new-Template-system-authorizing-Django-and-ezt-s.patch	6,01 ko	15 octobre 2017	Thomas Noël
0002-use-Django-ezt-templates-on-computed-expressions-194.patch	17,5 ko	15 octobre 2017	Thomas Noël
0003-use-Django-ezt-templates-in-messages-email-sms-journ.patch	32,8 ko	17 octobre 2017	Thomas Noël
0002-use-Django-ezt-templates-on-computed-expressions-194.patch	19,2 ko	17 octobre 2017	Thomas Noël
0001-add-new-Template-system-authorizing-Django-and-ezt-s.patch	6,01 ko	17 octobre 2017	Thomas Noël
0001-add-new-Template-system-authorizing-Django-and-ezt-s.patch	6,01 ko	18 novembre 2017	Thomas Noël
0002-use-Django-ezt-templates-on-computed-expressions-194.patch	19,2 ko	18 novembre 2017	Thomas Noël
0003-use-Django-ezt-templates-in-messages-email-sms-journ.patch	32,8 ko	18 novembre 2017	Thomas Noël
0003-use-Django-ezt-templates-in-messages-email-sms-journ.patch	32,8 ko	27 novembre 2017	Thomas Noël
0002-use-Django-ezt-templates-on-computed-expressions-194.patch	19,2 ko	27 novembre 2017	Thomas Noël
0001-add-new-Template-system-authorizing-Django-and-ezt-s.patch	6,02 ko	27 novembre 2017	Thomas Noël
0001-use-new-custom_template_email-system-19422.patch	2,04 ko	27 novembre 2017	Thomas Noël
0006-tests-add-some-Django-ezt-templates-tests-19442.patch	2,21 ko	27 novembre 2017	Thomas Noël
0005-workflows-allow-Django-formatted-receipient-in-sendma.patch	1,57 ko	27 novembre 2017	Thomas Noël
0004-misc-use-Django-ezt-templates-in-variadic-URLs-19442.patch	22,9 ko	27 novembre 2017	Thomas Noël