

## w.c.s. - Bug #45390

### foo|add:bar plante si bar n'existe pas

23 juillet 2020 11:55 - Thomas Noël

<b>Statut:</b>	Nouveau	<b>Début:</b>	23 juillet 2020
<b>Priorité:</b>	Normal	<b>Echéance:</b>	
<b>Assigné à:</b>		<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>	Non		
<b>Description</b> En cherchant à évaluer {{ foo add:bar }} on arrive sur TemplateError: impossibilité de faire le rendu du gabarit Django : Failed lookup for key [bar] in '[{\True\: True, \False\: ....  Il faudrait dans ce cas considérer que bar est None ou 0			

#### Historique

##### #1 - 23 juillet 2020 12:13 - Thomas Noël

En fait je pense qu'on ne peut rien faire : c'est natif dans Django de rechercher la valeur de droite dans le contexte.

##### #2 - 23 juillet 2020 14:59 - Paul Marillonnet

Thomas Noël a écrit :

En fait je pense qu'on ne peut rien faire : c'est natif dans Django de rechercher la valeur de droite dans le contexte.

Non, c'est une lacune de notre côté il me semble. Dans w.c.s. on essaie de tout faire en se basant sur `django.template.base.Variable`, même les expressions avec filtre.  
Dans Django natif c'est fait dans une classe à part, `django.template.base.FilterExpression`, dont la méthode `resolve` attrape les erreurs `VariableDoesNotExist` et poursuit son exécution en conséquence, notamment en tentant une substitution de l'expression par l'option de configuration du moteur de rendu [string\\_if\\_invalid](#).

##### #3 - 23 juillet 2020 15:14 - Paul Marillonnet

Paul Marillonnet a écrit :

Thomas Noël a écrit :

En fait je pense qu'on ne peut rien faire : c'est natif dans Django de rechercher la valeur de droite dans le contexte.

Non, c'est une lacune de notre côté il me semble. Dans w.c.s. on essaie de tout faire en se basant sur `django.template.base.Variable`, même les expressions avec filtre.

Dans Django natif c'est fait dans une classe à part, `django.template.base.FilterExpression`, dont la méthode `resolve` attrape les erreurs `VariableDoesNotExist` et poursuit son exécution en conséquence, notamment en tentant une substitution de l'expression par l'option de configuration du moteur de rendu [string\\_if\\_invalid](#).

Et non, analyse trop rapide. La levée d'erreur semble en fait tout à fait délibérée, c'est même ici précisément, le raises=True : <https://git.entrouvert.org/wcs/git/tree/wcs/qommon/form.py#n2537>

##### #4 - 23 juillet 2020 16:42 - Thomas Noël

en Django "natif" :

```
>>> from django.template import Template, Context
>>> Template('{{ foo|add:bar }}').render(Context({}))
...
VariableDoesNotExist: Failed lookup for key [bar] in "[{\True\: True, 'False\: False, 'None\: None}, {}]"
```

C'est avant l'exécution du filtre que bar est recherchée (pour appeler le filtre) et `VariableDoesNotExist` provoque un crash avant même l'entrée dans le filtre donc.

Je ne vois pas de solution évidente.

**#5 - 23 juillet 2020 17:43 - Paul Marillonnet**

Thomas Noël a écrit :

Je ne vois pas de solution évidente.

Ok, à vue de nez en changeant la valeur du raises pour False dans l'appel au rendu de gabarit dans w.c.s. alors l'expression entière serait remplacée par la chaîne de substitution en cas d'invalidité. Mais ce n'est clairement pas l'objet du ticket, i.e. que l'expression soit correctement évaluée avec bar à 0.