

## w.c.s. - Development #43651

### wipe(), ne pas autoriser les appels depuis une instance

05 juin 2020 09:02 - Frédéric Péters

<b>Statut:</b>	Fermé	<b>Début:</b>	05 juin 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 éviter les erreurs façon #43635#note-5, où .wipe() est appelé plutôt que .remove_self().			

#### Révisions associées

##### Révision 13a54d9f - 14 octobre 2022 11:09 - Frédéric Péters

misc: make sure .wipe() is only called on classes (#43651)

#### Historique

##### #1 - 30 septembre 2022 13:31 - Frédéric Péters

- Fichier 0001-misc-make-sure-.wipe-is-only-called-on-classes-43651.patch ajouté
- Statut changé de Nouveau à En cours
- Assigné à mis à Frédéric Péters
- Patch proposed changé de Non à Oui

Très basiquement ça serait

```
@classmethod
def wipe(cls):
+     assert isinstance(cls, type)
```

je ne sais pas ce qui avait pu me bloquer au moment de ce ticket.

J'attache déjà mon idée de patch mais je ne suis pas encore en mesure de pousser la branche pour jenkins, je ferai ça quand j'aurai du réseau adéquat.

##### #2 - 30 septembre 2022 15:41 - Frédéric Péters

- Patch proposed changé de Oui à Non

C'est qu'en fait on a beau appeler sur un objet, une fois dans la méthode le cls est bien la classe et pas l'objet.

##### #3 - 10 octobre 2022 14:08 - Frédéric Péters

J'ai quelque chose dans la branche mais c'est une catastrophe pour pylint (1352 warnings, avec principalement marqués tous les appels .wipe() des tests, ignorer globalement le warning en question je ne suis pas fan, mais taper 1300 # noqa non plus).

##### #4 - 10 octobre 2022 14:47 - Benjamin Dauvergne

La version dans Django hérite de classmethod, ça devrait aider (ou l'utiliser directement si elle est dispo depuis longtemps):

```
django$ git grep -A5 classonlymethod
django/utils/decorators.py:class classonlymethod(classmethod):
django/utils/decorators.py:    def __get__(self, instance, cls=None):
django/utils/decorators.py:        if instance is not None:
django/utils/decorators.py:            raise AttributeError("This method is available only on the class, not o
n instances.")
django/utils/decorators.py:        return super().__get__(instance, cls)
django/utils/decorators.py-
```

##### #5 - 11 octobre 2022 11:50 - Frédéric Péters

- Fichier 0001-misc-make-sure-.wipe-is-only-called-on-classes-43651.patch ajouté
- Statut changé de En cours à Solution proposée
- Patch proposed changé de Non à Oui

Voilà avec la version django de classonlymethod et pylint la vit bien.

#### #6 - 11 octobre 2022 11:59 - Lauréline Guérin

- Statut changé de Solution proposée à Solution validée

#### #7 - 14 octobre 2022 11:10 - Frédéric Péters

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

```
commit 13a54d9f7d841eec7f873f6e84d043aaf6858a42
Author: Frédéric Péters <fpeters@entrouvert.com>
Date: Fri Sep 30 13:26:45 2022 +0200
```

```
misc: make sure .wipe() is only called on classes (#43651)
```

#### #8 - 18 octobre 2022 12:15 - Transition automatique

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

#### #9 - 18 décembre 2022 04:41 - Transition automatique

Automatic expiration

### Fichiers

---

0001-misc-make-sure-.wipe-is-only-called-on-classes-43651.patch	1,49 ko	30 septembre 2022	Frédéric Péters
0001-misc-make-sure-.wipe-is-only-called-on-classes-43651.patch	3,67 ko	11 octobre 2022	Frédéric Péters