

## w.c.s. - Development #66620

### utiliser une table parallèle pour les "magictoken"

25 juin 2022 12:38 - Frédéric Péters

<b>Statut:</b>	Fermé	<b>Début:</b>	25 juin 2022
<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>			
Lors de la saisie d'une demande les infos de celles-ci sont associées à un "magictoken", qui est conservé/passé dans les pages, pour en fin de saisie récupérer les infos et les enregistrer dans une demande. Le grand plan dans <a href="#">#24635#note-6</a> serait de ne pas avoir ce stockage parallèle des données, de les avoir directement en brouillon dans la table formdata. Mais je n'ai jamais avancé sur ce plan parce qu'il y a plein de subtilités à gérer et pas assez de temps :/			
Une autre approche serait possible, en restant plus proche de la logique actuelle, en séparant juste la partie "magictokens" de la table des sessions (en se préoccupant uniquement de l'SQL).			
<b>Demandes liées:</b>			
Lié à w.c.s. - Bug #65004: Ticket chapeau champs obligatoires vides à la soum...		<b>Fermé</b>	<b>09 mai 2022</b>

#### Révisions associées

##### Révision 844617f9 - 29 juin 2022 07:56 - Frédéric Péters

sql: store transient data in a specific table (#66620)

#### Historique

##### #1 - 25 juin 2022 12:38 - Frédéric Péters

- Lié à Bug #65004: Ticket chapeau champs obligatoires vides à la soumission ajouté

##### #2 - 26 juin 2022 09:30 - Frédéric Péters

- Fichier 0001-sql-store-transient-data-in-a-specific-table-66620.patch ajouté

- Statut changé de Nouveau à Solution proposée

- Patch proposed changé de Non à Oui

Voilà ça se passe avec la modification de la classe sql.Session pour implémenter ses propres méthodes add\_magictoken, get\_by\_magictoken, remove\_magictoken, qui plutôt que se baser sur un dictionnaire magictokens stocké dans la session se basent sur des infos d'une table transient\_data (id, session\_id, data, last\_update\_time) posée sur le côté.

Par rapport au problème de perte de données, ici on charge/enregistre la donnée "magictoken" uniquement dans les pages qui font appel à magictoken (= pas de risque qu'une session dans un autre onglet sur une autre page / demande écrase quoique ce soit); et les accès parallèles qui accèdent au magictoken, ce sont autosave/live, qui ont ignore\_session=True, qui ne doivent déjà plus être le problème.

##### #3 - 27 juin 2022 11:51 - Lauréline Guérin

```
def add_magictoken(self, token, data):
    super().add_magictoken(token, data) # 1
    if not self.id:
        self.store() # 2
```

(1) le super fait:

```
if not self.magictokens:
    self.magictokens = {}
self.magictokens[token] = data
```

(2) et le store de Session fait:

```
# store transient data
```

```
for v in (self.magictokens or {}).values():
    v.store()
```

On ne risque pas de se retrouver à faire un store sur un dict ?

#### #4 - 27 juin 2022 12:47 - Frédéric Péters

En effet et les tests ne passent jamais dans ce if not self.id que j'ai posé trop théoriquement, j'ai modifié la branche pour faire

```
+ def add_magictoken(self, token, data):
+     assert self.id
+     super().add_magictoken(token, data)
+     self.magictokens[token] = TransientData(id=token, session_id=self.id, data=data)
+     self.magictokens[token].store()
```

#### #5 - 27 juin 2022 12:56 - Benjamin Dauvergne

```
'''CREATE TABLE %s (id varchar PRIMARY KEY,
                    session_id varchar,
```

Une fois n'est pas coutume dans w.c.s., je pense qu'on peut se permettre d'utiliser une clé étrangère, i.e remplacer la déclaration de la colonne session\_id par

```
session_id VARCHAR REFERENCES sessions(id) ON DELETE CASCADE
```

ça évitera d'avoir à gérer un nettoyage explicite.

#### #6 - 28 juin 2022 18:47 - Frédéric Péters

Avec cette modif CASCADE + un test supplémentaire test\_transient\_data\_removal.

#### #7 - 28 juin 2022 18:47 - Frédéric Péters

- Fichier 0001-sql-store-transient-data-in-a-specific-table-66620.patch ajouté

#### #8 - 28 juin 2022 22:03 - Benjamin Dauvergne

- Statut changé de Solution proposée à Solution validée

Ok, faut que ça vive en recette.

#### #9 - 29 juin 2022 07:57 - Frédéric Péters

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

(je tag/déploie une fois le build passé)

```
commit 844617f9330aa71bf71e9a4498c36da44b64ed69
Author: Frédéric Péters <fpeters@entrouvert.com>
Date: Sat Jun 25 11:16:02 2022 +0200
```

```
sql: store transient data in a specific table (#66620)
```

#### #10 - 29 juin 2022 10:14 - Transition automatique

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

#### #11 - 04 septembre 2022 04:42 - Transition automatique

Automatic expiration

### Fichiers

0001-sql-store-transient-data-in-a-specific-table-66620.patch	11,7 ko	26 juin 2022	Frédéric Péters
0001-sql-store-transient-data-in-a-specific-table-66620.patch	12,3 ko	28 juin 2022	Frédéric Péters