

## w.c.s. - Development #81183

### Revoir le stockage et la gestion des "visited\_object"

13 septembre 2023 19:20 - Frédéric Péters

<b>Statut:</b>	En cours	<b>Début:</b>	28 août 2023
<b>Priorité:</b>	Normal	<b>Echéance:</b>	
<b>Assigné à:</b>	Pierre Ducroquet	<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>			
<p>(copie de <a href="#">#80613</a> qui a été associé à un autre commit, plus simple, aussi je retire la priorité "haute" vu que ça ne semble plus le cas).</p> <p>Actuellement, l'information sur les visited_object "squatte" le stockage des sessions. Cela a conduit ce matin à avoir la requête suivante qui a consommé un temps CPU considérable sur le serveur (d'autant plus que nos logs tronquent à 100ms, j'ignore donc le temps réellement utilisé en permanence par cette requête)</p> <pre>select id, session_data from sessions where last_update_time &gt;= ?::timestamp;</pre> <p>Ce qui prend du temps n'est pas le filtrage, mais le décodage côté PG des données pour les renvoyer à l'application, l'encodage réseau, le temps réseau, puis le temps de décodage côté applicatif. Par ailleurs, autre facteur à prendre en compte : le stockage dans le session_data, un bytea contenant un objet pickle, est particulièrement inefficace à la mise à jour puisque le code python doit tout reconstruire, et cela fait des enregistrements particulièrement volumineux pour peu de données utiles.</p> <p>J'ai identifié le code suivant comme fort possiblement responsable de cet appel:</p> <pre>pierre@entrouvert-pierred:~/eo/wcs\$ git grep select_recent wcs/sessions.py:         for session in cls.select_recent(ignore_errors=True): wcs/sessions.py:         for session in self.__class__.select_recent(ignore_errors=True): wcs/sql.py:     def select_recent(cls, seconds=30 * 60, **kwargs):</pre> <p>Cela correspond aux méthodes Session::get_visited_objects et Session::unmark_visited_object. La seconde prend en paramètre un formdata, elle sait donc quel objet doit être retiré. Un stockage efficace doit prendre en compte ce besoin. La première renvoie une liste de tous les objets visités, en en excluant éventuellement un utilisateur.</p> <p>Je suggère de remplacer tout ceci par une table "visited_object" (quelle imagination) avec la définition suivante:</p> <pre>CREATE TABLE visited_object (   object_key character varying not null,   visit_timestamp timestamp with time zone not null,   visitor integer );</pre> <p>Je n'ai pas l'impression que l'on puisse imposer une unicité au sein de cette table (et inutile de créer une fausse clé primaire sans intérêt).</p>			
<b>Demandes liées:</b>			
Lié à w.c.s. - Development #80613: Revoir le stockage et la gestion des "visi...		Fermé	28 août 2023

#### Historique

##### #1 - 13 septembre 2023 19:20 - Frédéric Péters

- Lié à Development #80613: Revoir le stockage et la gestion des "visited\_object" ajouté

##### #2 - 13 septembre 2023 19:21 - Robot Gitea

- Statut changé de Nouveau à En cours

Pierre Ducroquet (pducroquet) a commencé à travailler sur une pull request sur Gitea concernant cette demande :

- URL : <https://git.entrouvert.org/entrouvert/wcs/pulls/624>
- Titre : WIP: create a VisitedObjects class and table to reduce Sessions table usage (#81183)
- Modifications : <https://git.entrouvert.org/entrouvert/wcs/pulls/624/files>