

w.c.s. - Bug #12534

performances SQL vue globale

11 juillet 2016 22:27 - Frédéric Péters

Statut:	Fermé	Début:	11 juillet 2016
Priorité:	Haut	Echéance:	
Assigné à:	Frédéric Péters	% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:	v1.52	Planning:	
Patch proposed:	Oui		
Description			
<pre>t0 = time.time() criterias = [] criterias.append(NotEqual('status', 'draft')) criterias.append(Equal('is_at_endpoint', False)) criterias.append(Intersects('actions_roles_array', user_roles)) formdatas = sql.AnyFormData.select(criterias, order_by='receipt_time', limit=20, offset=0) print [x.id for x in formdatas], time.time()-t0 → 5.49 secondes vs t0 = time.time() formdatas = [] for formdef in FormDef.select(): criterias = [] criterias.append(Intersects('actions_roles_array', user_roles)) endpoint_status = ['wf-%s' % x.id for x in formdef.workflow.get_endpoint_status()] criterias.append(NotContains('status', ['draft'] + endpoint_status)) formdatas.extend(formdef.data_class().select(criterias)) formdatas.sort(key=lambda x: getattr(x, 'receipt_time')) print [x.id for x in formdatas][:20], time.time()-t0 → 0.29 secondes [...]</pre>			
Après ajout d'index sur *_evolutions(time) et *_evolutions(formdata_id), la première forme prend 1.47 secondes.			

Révisions associées

Révision 1aa890c8 - 13 juillet 2016 10:24 - Frédéric Péters

sql: store computed last_update_time in table (#12534)

Even with indices the last_updated_time subselect are too slow for the global view and statistics.

Révision 611fdf3c - 13 juillet 2016 10:24 - Frédéric Péters

sql: drop materialized view support (#12534)

It's no longer necessary as requests on wcs_all_forms are now as fast.

Historique

#1 - 11 juillet 2016 22:28 - Frédéric Péters

Vague diff de l'ajout des index :

```
@@ -416,6 +416,23 @@ def do_formdef_tables(formdef, conn=None, cur=None, rebuild_views=False, rebuild
    # them even if not asked to.
    redo_views(conn, cur, formdef, rebuild_global_views=rebuild_global_views)
+
+ conn.commit()
+
```

```

+   try:
+       cur.execute('CREATE INDEX %s_time_idx ON %s_evolution(time)' % (
+           table_name, table_name))
+       conn.commit()
+   except psycopg2.ProgrammingError:
+       conn.rollback()
+
+   try:
+       cur.execute('CREATE INDEX %s_fid_idx ON %s_evolution(formdata_id)' % (
+           table_name, table_name))
+       conn.commit()
+   except psycopg2.ProgrammingError:
+       conn.rollback()
+
+

```

#2 - 11 juillet 2016 22:37 - Frédéric Péters

Détail du temps passé dans la vue d'affichage de la vue globale, **après création des index** :

```

0.0000 -- start
0.0287 -- before count
1.5234 -- after count
3.0185 -- after select
3.0187 -- before getting visited objects
5.1532 -- after getting visited objects
5.2788 -- after formdata loop
5.3371 -- end

```

→ point supplémentaire, l'inspection des sessions, pour réduire le temps passé à découvrir les objets "verrouillés".

4851 sessions.

→ exécution manuelle du job "clean_sessions"

148 sessions.

→ **trouver pourquoi le job ne s'exécute pas.**

Nouveaux comptes :

```

0.0000 -- start
0.0107 -- before count
1.4957 -- after count
2.9818 -- after select
2.9820 -- before getting visited objects
3.0309 -- after getting visited objects
3.1563 -- after formdata loop
3.2131 -- end

```

#3 - 12 juillet 2016 09:26 - Frédéric Péters

- *Sujet changé de performances vue globale à performances SQL vue globale*

- *Assigné à mis à Frédéric Péters*

#4 - 12 juillet 2016 10:55 - Frédéric Péters

- *Fichier 0002-sql-drop-materialized-view-support-12534.patch ajouté*

- *Fichier 0001-sql-store-computed-last_update_time-in-table-12534.patch ajouté*

- *Statut changé de Nouveau à En cours*

- *Patch proposed changé de Non à Oui*

Ajouter un index sur *evolution(formdata_id, time) améliore encore sensiblement les performances, particulièrement sur mon jeu de données de test, moins sur ma copie des données d'Alfortville.

Du coup je prends une option différente, je stocke dans les tables le last_update_time, plutôt que le calculer via un subSELECT. Et comme les perms sont désormais présentes, ça permet de virer le code de gestion des vues matérialisées.

Ça demande un premier redémarrage un peu lent, parce qu'il doit faire un .store() sur tous les formdata. Sur mes données locales, qui ont vu les années passer, j'ai eu un crash sur le store d'une donnée incompatible, je n'ai pour autant pas mis de try/except autour du .store(), ce qui fait que le redémarrage peut échouer. (ça se discute mais je pense que si ça foire sur la prod, ça révélera un vrai problème, plutôt que le glisser sous le tapis).

#5 - 12 juillet 2016 10:55 - Frédéric Péters

- *Priorité changé de Normal à Haut*

#6 - 12 juillet 2016 11:02 - Frédéric Péters

- *Fichier 0002-sql-drop-materialized-view-support-12534.patch ajouté*

Suppression de l'appel toutes les heures au rafraichissement des vues matérialisées.

#7 - 12 juillet 2016 11:17 - Frédéric Péters

Ça demande un premier redémarrage un peu lent, parce qu'il doit faire un `.store()` sur tous les formdata. Sur mes données locales, qui ont vu les années passer, j'ai eu un crash sur le store d'une donnée incompatible, je n'ai pour autant pas mis de try/except autour du `.store()`, ce qui fait que le redémarrage peut échouer. (ça se discute mais je pense que si ça foire sur la prod, ça révélera un vrai problème, plutôt que le glisser sous le tapis).

Du coup je viens de faire ça sur la recette et la prod, déjà aucun crash. Et comme info supplémentaire, pour la prod, le temps pour les différents vhost, c'est ~2 minutes.

#8 - 13 juillet 2016 10:24 - Thomas Noël

Ack

#9 - 13 juillet 2016 10:26 - Frédéric Péters

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

```
commit 611fdf3cbb44b547380d28b94cddf029671773e7
Author: Frédéric Péters <fpeters@entrouvert.com>
Date: Tue Jul 12 10:45:05 2016 +0200
```

```
sql: drop materialized view support (#12534)
```

```
It's no longer necessary as requests on wcs_all_forms are now as fast.
```

```
commit 1aa890c826641d88c1032f5f8183f59fec3c6cfa
Author: Frédéric Péters <fpeters@entrouvert.com>
Date: Tue Jul 12 09:27:14 2016 +0200
```

```
sql: store computed last_update_time in table (#12534)
```

```
Even with indices the last_updatet_ime subselect are too slow for the global view and statistics.
```

#10 - 04 décembre 2016 12:52 - Frédéric Péters

- *Version cible mis à v1.52*

#11 - 04 décembre 2016 13:12 - Frédéric Péters

- *Statut changé de Résolu (à déployer) à Fermé*

Fichiers

0002-sql-drop-materialized-view-support-12534.patch	5,21 ko	12 juillet 2016	Frédéric Péters
0001-sql-store-computed-last_update_time-in-table-12534.patch	10,1 ko	12 juillet 2016	Frédéric Péters
0002-sql-drop-materialized-view-support-12534.patch	6,25 ko	12 juillet 2016	Frédéric Péters