

Hobo - Development #34533

optimisation: rendre possible de ne pas exécuter les migrations sur la création d'un tenant

04 juillet 2019 11:34 - Benjamin Dauvergne

Statut:	Nouveau	Début:	04 juillet 2019
Priorité:	Normal	Echéance:	
Assigné à:		% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	Non
Patch proposed:	Non		

Description

Le but initial était d'accélérer mes tests sur les tenants dans hobo.

J'ai testé avec succès la désactivation des migrations sur la création d'un tenant en réutilisant la méthode de pytest-django :

```
--- tenant_schemas/models.py      2019-07-04 11:30:16.040058558 +0200
+++ tenant_schemas/models.py      2019-07-04 11:30:08.388007819 +0200
@@ -113,10 +113,19 @@
     cursor.execute('CREATE SCHEMA %s' % self.schema_name)

     if sync_schema:
+         from pytest_django.fixtures import _disable_native_migrations
+
+         sentinel = object()
+         MIGRATIONS_MODULES = getattr(settings, 'MIGRATION_MODULES', sentinel)
+         _disable_native_migrations()
+         call_command('migrate_schemas',
+                      schema_name=self.schema_name,
+                      interactive=False,
+                      run_syncdb=True,
+                      verbosity=verbosity)
+         if MIGRATIONS_MODULES is sentinel:
+             del settings.MIGRATIONS_MODULES
+         else:
+             settings.MIGRATIONS_MODULES = MIGRATIONS_MODULES

+         connection.set_schema_to_public()
```

Pour référence le code dans `django.db.backends.base.creation.BaseDatabaseCreation.create_test_db` qui fait que ça marche dans `pytest-django` (l'important c'est le `run_syncdb`) :

```
61         # We report migrate messages at one level lower than that requested.
62         # This ensures we don't get flooded with messages during testing
63         # (unless you really ask to be flooded).
64         call_command(
65             'migrate',
66             verbosity=max(verbosity - 1, 0),
67             interactive=False,
68             database=self.connection.alias,
69             run_syncdb=True,
70         )
```

Il faut pour cela que les applications ne créent pas de données dans leurs migrations (c'est de toute façon un abus, il faut utiliser soit des fixtures soit le signal `post_migrate` pour cela).

Le seul manque ici c'est que `migrate` avec `run_syncdb` ne lance pas les signaux `pre/post-migrate` qui par exemple sur `authentic` sont très importants (création des rôles de bas et de l'OU par défaut).

Historique

#1 - 04 juillet 2019 11:35 - Benjamin Dauvergne

- Description mis à jour