

w.c.s. - Development #57299

stocker un "diff" dans l'historisation, plutôt que l'objet complet

27 septembre 2021 14:33 - Frédéric Péters

Statut:	Fermé	Début:	27 septembre 2021
Priorité:	Normal	Echéance:	
Assigné à:	Lauréline Guérin	% réalisé:	0%
Catégorie:		Temps estimé:	0:00 heure
Version cible:		Planning:	Non
Patch proposed:	Oui		

Description

Dans l'historisation on enregistre l'export XML de l'objet, sauf si ça fait plus d'un Mo, parce que d'expérience ça correspond à des workflows avec des modèles de document intégrés et tout enregistrer systématiquement consomme trop vite l'espace disque de la base de données.

Mais on pourrait imaginer conserver un diff, entre la version en cours et la dernière version "complète", et uniquement de temps en temps (genre quand le diff devient plus gros que la sérialisation intégrale) enregistrer la version intégrale.

Révisions associées

Révision 0565ffe5 - 18 octobre 2021 15:27 - Lauréline Guérin

snapshot: store a patch instead of serialization (#57299)

Historique

#1 - 04 octobre 2021 14:52 - Lauréline Guérin

- Assigné à mis à Lauréline Guérin

#2 - 05 octobre 2021 14:19 - Lauréline Guérin

- Fichier 0001-snapshot-store-a-patch-instead-of-serialization-5729.patch ajouté

- Statut changé de Nouveau à Solution proposée

- Patch proposed changé de Non à Oui

proposition

difflib permet de générer un diff avec ndiff, mais qui contient tout (les lignes non modifiées, les mods), donc forcément plus volumineux que la serialization d'origine, mais à partir duquel en appliquant diffliib.restore on peut builder la version qu'on veut.

Mais c'est volumineux

difflib permet aussi de générer un diff plus au format patch (avec juste les mods) avec , plus intéressant ici. Mais rien pour appliquer ce patch. J'ai fouillé un peu partout et je suis tombée sur <https://gist.github.com/noporpoise/16e731849eb1231e86d78f9dfeca3abc> que j'ai copié/collé (avec 2/3 adaptations de style)

Ca marchouille.

#3 - 15 octobre 2021 14:22 - Frédéric Péters

```
# should we store a snapshot ?
store_snapshot = False
```

Comme c'est désormais il devient difficile de capter le moment où ça resterait à False; je pense que le code serait plus clair en inversant,

```
store_snapshot = True
if label is None:
    # compare with patch of latest snapshot
    latest = cls.get_latest(obj.object_type, obj.object_id)
    if latest.patch and patch == latest.patch:
        # the patch is the same as in the previous snapshot
        store_snapshot = False
```

(il me semble que c'est la situation)

#4 - 18 octobre 2021 10:16 - Lauréline Guérin

- Fichier 0001-snapshot-store-a-patch-instead-of-serialization-5729.patch ajouté

#5 - 18 octobre 2021 13:02 - Frédéric Péters

- Statut changé de Solution proposée à Solution validée

#6 - 18 octobre 2021 20:49 - Lauréline Guérin

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

```
commit 0565ffe5d3faccb89b80017cd872e47f8863e967
Author: Lauréline Guérin <zebuline@entrouvert.com>
Date: Tue Oct 5 11:38:28 2021 +0200
```

```
snapshot: store a patch instead of serialization (#57299)
```

#7 - 19 octobre 2021 18:17 - Frédéric Péters

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

Fichiers

0001-snapshot-store-a-patch-instead-of-serialization-5729.patch	19,7 ko	05 octobre 2021	Lauréline Guérin
0001-snapshot-store-a-patch-instead-of-serialization-5729.patch	19,4 ko	18 octobre 2021	Lauréline Guérin