w.c.s. - Bug #6927

avoir des dates UTC dans l'export json (et ailleurs ?)

10 avril 2015 01:07 - Thomas Noël

Statut:	Fermé	**Début:**	10 avril 2015
Priorité:	Normal	**Echéance:**	
Assigné à:	Thomas Noël	**% réalisé:**	0%
Catégorie:		**Temps estimé:**	0:00 heure
Version cible:			
Patch proposed:	Oui	**Planning:**	

Description

Dans l'export JSON, les dates sont locales et sans info de timezone.

Ca me pose des soucis dans elasticsearch (pour qui une date sans timezone est en UTC, c'est assez logique). Ça en posera pour d'autres connexion sans doute.

Je n'ai rien trouvé de propre pour transformer un time.localtime() en UTC ou y ajouter l'info de timezone.

Révisions associées

Révision 3c6b7200 - 17 avril 2015 14:56 - Thomas Noël

json: export DateField as YYYY-MM-DD (#6927)

Révision 2ba78294 - 17 avril 2015 14:58 - Thomas Noël

json: always use UTC datetime format (#6927)

Historique

#1 - 11 avril 2015 23:14 - Frédéric Péters

En utilisant pytz.timezone('Europe/Paris').localize(datetime); à noter que ça veut aussi dire avoir un paramétrage quelque part pour la timezone ou alors, sans doute plus simplement, prendre la config du serveur (mais c'est quelque chose qui varie de distrib en distrib, en attendant systemd, à voir…).

Dans create_publisher il y aurait donc un appel cls.load_local_timezone, qui irait (dans debian) lire /etc/timezone, et ajouterait au publisher un attribut timezone; dans le JSONEncoder, le time.struct_time serait transformé en datetime, passé dans get_publisher().timezone.localize et retourné avec un '%Z' dans sa chaine de formatage.

À noter qu'à l'occasion de ce travail, il pourrait y avoir suppression du qommon/strftime.py, dans la mesure où en passant par des objets datetime il sera possible d'avoir un strftime fonctionnant avec des dates < 1900.

#2 - 12 avril 2015 12:35 - Thomas Noël

Benjamin m'a fait une démonstration à base de mktime, gmtime, calendar, mais j'avoue ne pas parvenir à la refaire. Benj, tu peux le redire ici ?

#3 - 12 avril 2015 23:20 - Benjamin Dauvergne

Ca suppose que la timezone soit constante dans le temps sur la machine.

```
from time import gmtime, mktime

def localtime_to_gmtime(t):
    return gmtime(mktime(t))
```

Le mieux ce serait quand même de stocker des datetime timezone aware et toujours dans UTC. Et pour l'export je dirai qu'on se fout de savoir si c'est en UTC mais il faut que ce soit une chaîne ISO8601 avec toujours une indication de timezone.

#4 - 12 avril 2015 23:37 - Benjamin Dauvergne

```
datetime.datetime.utcfromtimestamp(time.mktime(t)).isoformat()+'Z'
```

#5 - 14 avril 2015 14:44 - Thomas Noël

... et de fait, on le fait déjà pour le stockage SQL: http://git.entrouvert.org/wcs.git/tree/wcs/sql.py#n874

#6 - 14 avril 2015 14:54 - Thomas Noël

- Fichier 0001-json-always-use-UTC-datetime-format-6927.patch ajouté

- Statut changé de Nouveau à En cours

- Assigné à mis à Thomas Noël

- Patch proposed changé de Non à Oui

Voici donc une proposition de patch, où toutes les dates JSON seront en exprimées UTC (avec un 'Z' final)

C'est "violent" car ça va modifier tous nos exports JSON, mais je pense qu'on ne gère pas encore vraiment les dates dans nos utilisations de l'API actuelle ; je suis donc partisan de faire ce changement maintenant.

PS: je ne sais pas encore comment écrire un test (il faut imposer la TZ, je regarde).

#7 - 16 avril 2015 14:32 - Thomas Noël

- Fichier 0001-json-export-DateField-as-YYYY-MM-DD-6927.patch ajouté

En écrivant les tests, bogue : parce qu'une date dans wcs c'est juste un jour, et on se retrouve avec 2013-10-10T00:00 -> 2013-10-09T22:00:00Z en été à Paris...

Ce patch permet de gérer correctement les dates exportés dans le dictionnaire *fields*. Les autres datetime (receipt_time, last_update_time), avec le patch précédent, seraient bien exportées en UTC.

#8 - 16 avril 2015 18:03 - Frédéric Péters

> Voici donc une proposition de patch, où toutes les dates JSON seront en exprimées UTC (avec un 'Z' final)

Je trouverais plutôt mieux que l'info dans le json soit la date/heure mise en forme sur la timezone locale (mais ça demande sans doute de passer par pytz, peut-être même aussi à tzlocal, ou peut-être le LocalTimezone de la page https://docs.python.org/2/library/datetime.html).

#9 - 16 avril 2015 18:13 - Thomas Noël

L'ajout de la timezone ne me parait pas une bonne idée pour ce qui concerne les API (ça complexifie le traitement côté consommateur). Et quand je vois ce qu'il faudra faire, pytz ou équivalent, bof...

#10 - 17 avril 2015 14:43 - Frédéric Péters

Ok.

#11 - 17 avril 2015 14:59 - Thomas Noël

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

C'est poussé, avec d'abord l'export YYYY-MM-DD pour les champs date, puis le patch "UTC" pour les time.struct_time.

```
commit 2ba78294e4c6c1a6943ec64aa00058e6d74f08e8
Author: Thomas NOEL <tnoel@entrouvert.com>
Date:   Tue Apr 14 14:51:18 2015 +0200

    json: always use UTC datetime format (#6927)

commit 3c6b72006f58352a088efa398e105c50f8f41ce5
Author: Thomas NOEL <tnoel@entrouvert.com>
Date:   Thu Apr 16 12:32:46 2015 +0200

    json: export DateField as YYYY-MM-DD (#6927)
```

#12 - 26 août 2015 14:37 - Thomas Noël

- Statut changé de Résolu (à déployer) à Fermé

Fichiers

0001-json-always-use-UTC-datetime-format-6927.patch	880 octets	14 avril 2015	Thomas Noël
0001-json-export-DateField-as-YYYY-MM-DD-6927.patch	2,07 ko	16 avril 2015	Thomas Noël