

## conf-scribe - Anomalie #831

### Erreur au démarrage de cups : cherche a charger lp

19/08/2010 14:41 - Emmanuel GARETTE

<b>Statut:</b>	Fermé	<b>Début:</b>	19/08/2010
<b>Priorité:</b>	Normal	<b>Echéance:</b>	
<b>Assigné à:</b>		<b>% réalisé:</b>	100%
<b>Catégorie:</b>		<b>Temps estimé:</b>	0.00 heure
<b>Version cible:</b>	2.3 BETA 1	<b>Temps passé:</b>	0.00 heure
<b>Distribution:</b>			

#### Description

Le script de démarrage de CUPS cherche a chargé le module noyau lp.

Ce module n'est évidemment pas présent dans le conteneur, cela génère un message d'erreur.

#### Révisions associées

##### Révision 0bd0ddd7 - 12/04/2006 00:53 - mithrandi

Merge json-unicode-surrogates-831-2.

Fixes #831

Author: mithrandi

Reviewer: exarkun, glyph

This branch changes nevw.json's string encoding procedure to minimize the amount of escaping that is done. Per the JSON Working Draft 04:

```
All Unicode characters may be placed within
the quotation marks except for the characters which must be
escaped: quotation mark, reverse solidus, and the control
characters (U+0000 through U+001F).
```

JSON is defined as a text format, the encoding in transit being determined by the protocols / implementations in use. An intentional side-effect of this change is that unicode codepoints above U+FFFF can now be encoded on Python builds with 32-bit unicode storage; when decoded in a JavaScript interpreter in a browser, they will typically be represented by the relevant UTF-16 surrogate pair.

As it turns out, this was already sort-of working on 16-bit unicode Python builds, by accident: on these builds, the surrogate pair used to represent the high codepoint is exposed to user code, and thus would have been encoded directly as a surrogate pair, which when decoded would result in the same string. As the Python UTF-\* codecs behave correctly in the presence of surrogate pairs in the original string, and Nevow currently always uses UTF-8 as the HTTP encoding anyway, 16-bit unicode builds now encode strings containing these high codepoints in the same way as their 32-bit unicode counterparts.

There is one remaining defect; another implementation could elect to encode a high codepoint as an escaped surrogate pair instead of embedding the character directly; that is, "\uD834\uDD1E" instead of "\U0001D11E". In this case, a 32-bit Python unicode build will receive a unicode string with the surrogate pair embedded on decoding, which is unexpected. 16-bit unicode builds remain unaffected, because the surrogates would appear in the string for them under either circumstance. As MochiKit.Base.serializeJSON does not escape characters in this fashion at all (it always embeds the codepoint directly), this is not a problem for Athena.

Another nice effect of this change is that string encoding is simpler, faster, and results in smaller encoded strings.

**Révision a030869b - 19/08/2010 14:42 - Emmanuel Garette**

ajout d'une question pour charge le module LP de cups (fixes #831)

**Révision 3ba8baf9 - 25/07/2016 18:00 - Alexandre Delaunay**

prevent projects list to display same project several time; fix #831

## Historique

---

**#1 - 25/08/2010 11:35 - Emmanuel GARETTE**

- Statut changé de Nouveau à Résolu

- % réalisé changé de 0 à 100

Appliqué par commit [a030869b5952e59ad9be66d41ad528d7222427a8](#).

**#2 - 16/09/2010 14:40 - Joël Cuissinat**

- Statut changé de Résolu à Fermé