

Shoop - Bug # 139: Problème avec l'argument last du backend

Status:	Assigné	Priority:	Normal
Author:	David THENON	Category:	Interface web
Created:	07/23/2009	Assigned to:	David THENON
Updated:	11/16/2009	Due date:	
Subject:	Problème avec l'argument last du backend		
Description:	Par "Moltonelle epaisseur triple": <pre> sveetch< j'ai 2 ptits bugs à te remonter concernant le paramètre last= dans les requêtes backend 23:26:03 Le premier, c'est que, quand le paramètre last est spécifié, les posts ne sont pas remontés dans le bon ordre dans le backend (l'usage veut que ce soit par id décroissant, ce qui est fait correctement sans le last) 23:26:03 Le second, c'est que si l'on met le paramètre last à une valeur inférieure à [dernier id dans la base]-[taille max du backend (=84)] alors il remonte les 84 premiers posts à partir de ce numéro. Il faudrait plutôt qu'il remonte les 84 derniers posts en base </pre>		

History

07/23/2009 02:10 PM - David THENON

En fait je comprends pas surtout la seconde partie.

Autre chose, tu fais ça sur le backend XML ou JSON ? Parce qu'ils ont pas le même comportement. Parce que pour le backend XML je dois être enclin à changer ça sans soucis vu que wmcc ne gère pas l'argument "last". Mais je crois me rappeler que j'avais calé le comportement de ce backend pour convenir à Pycc, donc ça m'étonne un peu ta demande.

07/23/2009 02:39 PM - Chrisix Moltonelle

Petits éclaircissements, tout d'abord il s'agit bien de la version XML dont je parle.

Pour le point 1) il suffit de trier les posts par id décroissant comme c'est le cas par défaut (sans le last).

Pour le point 2) je prends un exemple : mettons que, en base, l'id du post le plus récent soit 4200. Mettons maintenant que je fasse une requête backend avec en paramètre last=4000. Si le backend xml n'était pas limité en taille, on devrait recevoir les posts n°4001 à 4200. Mais bien entendu, il y a une taille maximale, de 84 posts ici. Le serveur devrait donc renvoyer les posts n°4117 à 4200, or il renvoie les posts n°4001 à 4084.

07/23/2009 03:36 PM - David THENON

```
<pre>  
15:21:48 Moltonelle epaisseur triple 15:19:04 soit r = l'id du post le plus récent en base, t = taille max du backend  
et l = l'id last passé en paramètre de la requête, alors le serveur doit renvoyer les max((r-l),t) posts les plus récents  
</pre>
```

```
<pre>  
15:29:25 Moltonelle epaisseur triple 15:21:23 Il faut garder à l'esprit que quand je fais une requête backend, je veux  
les posts les plus récents, actuels. Si je la fais avec last=4000, c'est parce que je veux des posts d'id minimal 4000,  
pas parce que je veux l'historique des posts dont l'id commence à 4000. Si je veux un historique, j'utiliserai la fonction  
d'historique, pas une requête backend  
</pre>
```

```
<pre>
```

15:31:43 Moltonelle epaisseur triple 15:29:25 maintenant si dans la base, l'id le plus récent est 4042, très bien, j'aurai les 42 posts les plus récents. Si l'id le plus récent est 4200 ben c'est que mon client a °bloubé° pour une raison X (fréquence de rafraichissement trop faible, connexion blo...) et tant pis s'il a paumé des posts au passage
</pre>

07/23/2009 03:51 PM - Chrisix Moltonelle

Erreur dans ma formule d'ailleurs, c'est le plus petit des deux nombres (r-l) et t qu'il faut prendre bien entendu :-)

11/16/2009 02:53 PM - Aurélien DEHAY

ça avance pas vite cette histoire!

En théorie, pour les 84 derniers/premiers a remonter, il suffit en SQL d'inverser le order by.