

SPIP - Evolution #3899

Comportement des inclusions avec le paramètre connect.

02/10/2017 02:37 PM - marcimat ☐☐

Status:	Fermé	Start date:	02/10/2017
Priority:	Normal	Due date:	
Assignee:		% Done:	0%
Category:		Estimated time:	0.00 hour
Target version:	3.2		
Resolution:	fixed		

Description

Il y a un comportement contre-intuitif dans un cas d'usage des inclusions et du paramètre connect, ce qu'à révélé une petite analyse dans [#3823](#).

J'en fais un ticket dédié car c'est un problème distinct de ce qui est soulevé là bas.

Fonctionnement actuel

Le paramètre d'URL connect

Le paramètre connect=truc dans une URL d'un site SPIP permet d'indiquer à SPIP qu'il doit utiliser le connecteur SQL 'truc' dans les squelettes et les boucles utilisés, et donc utiliser config/truc.php en lieu et place de config/connect.php. À différents endroits du compilateur, ce paramètre est séparé du contexte d'environnement du squelette, notamment dans les boucles où il atterrit dans \$boucles[\$idb]->sql_serveur et sera utilisé pour les requêtes SQL générées.

Inclusions en indiquant un connect spécifique

Une autre manière d'indiquer que les squelettes / boucles utilisent un connecteur spécifique est de transmettre l'environnement connect=truc aux inclusions, de la sorte :

```
<INCLUDE{fond=test, connect=truc} />
#INCLUDE{fond=test, connect=truc}
```

Cette option d'inclusion est prise en compte dans recuperer_fond().

Inclusion connect=truc et paramètre d'url connect=bidule

Lorsqu'on a à la fois dans l'URL &connect=bidule, et une inclusion qui indique un connect spécifique tel que #INCLUDE{fond=test, connect=truc}

alors d'inclusion actuellement est chargée avec le paramètre d'URL, c'est à dire en utilisant connect=bidule.

C'est cela qui est assez contre-intuitif et logiquement non désiré (le connect forcé ici devrait prendre le pas sur celui d'environnement).

Comment améliorer ?

Où cela se passe dans le code ?

Ce qui se passe lorsqu'il y a cette écriture est, comme expliqué, que le connect dans l'URL reste prioritaire sur celui de l'argument transmis à #INCLUDE ou <INCLUDE>.

- Pour #INCLUDE ça se passe dans balise_INCLUDE_dist() là <https://core.spip.net/projects/spip/repository/entry/spip/ecrire/public/balises.php#L1995>
- Pour <INCLUDE> là ça se passe dans calculer_inclure() là

Solutions ?

À ces 2 endroits, il faudrait du coup prendre en compte en priorité le `$contexte['connect']` si existant

- soit dans les 2 appels à la constante `CODE_RECUPERER_FOND` plutôt que d'écrire directement `_request('connect')`, mais c'est un peu compliqué car c'est du code compilé
- soit directement dans `recuperer_fond()`, plus facile, inverser l'ordre :

```
if (isset($contexte['connect'])) {  
    $connect = ($connect ? $connect : $contexte['connect']);  
    unset($contexte['connect']);  
}
```

deviendrait :

```
if (isset($contexte['connect'])) {  
    $connect = $contexte['connect'];  
    unset($contexte['connect']);  
}
```

Cette dernière correction serait la plus simple et la plus facile. Ça permettrait même d'annuler un connect supérieur dans une inclusion en passant `{connect=""}`.

Des avis ?

Related issues:

Related to SPIP - Anomalie #3823: Pagination et connect

Fermé

08/25/2016

History

#1 - 02/10/2017 02:37 PM - marcimat

- Related to Anomalie #3823: Pagination et connect added

#2 - 02/10/2017 02:48 PM - b b

Merci pour cette analyse détaillé :)

Perso, je penche plus pour ta dernière proposition pour sa simplicité et le résultat fonctionnel obtenu.

#3 - 02/10/2017 08:07 PM - marcimat

<https://core.spip.net/projects/spip/repository/revisions/14024> pour info sur l'origine de ce code.

#4 - 02/11/2017 11:23 AM - marcimat

- Status changed from Nouveau to Fermé

- Resolution set to fixed

Appliqué en 3.2 par <https://core.spip.net/projects/spip/repository/revisions/23411>

#5 - 02/11/2017 11:33 AM - b b

Bien joué, grande classe :)