

Mögliche Anfragen

1 Reset

1.1 *index.php*

Berechne alle F_boxes[i] mit Treffer 1 bis \$AC->settings->mcs, mit query == "
H_boxes[1] und W_boxes[1] bleiben für diese leere Anfrage versteckt

1.2 --

2 Sprung von DBLP Trier nach dblp-mirror

2.1 *index.php?query=xxx&qt=F&qj=1&mcs=14&reset*

Berechne F_boxes[1] mit mcs=14

Berechne H_boxes[1] mit Treffer 1 bis \$AC->settings->hpp

Berechne W_boxes[1], F_boxes[2], F_boxes[3] mit Treffer 1 bis \$AC->settings->mcs

*Achtung: Muß unterschieden werden von 6.1, wo die URL bis auf den query-Parameter identisch ist.
Hier werden die übrigen Boxen berechnet, bei 6.1 werden sie aus \$_SESSION übernommen.*

Geeignete Methode, um 2.1 von 6.1 zu unterscheiden, steht noch aus.

Alternativen, die wir z.Z. Prüfen:

- *X_boxes == null (führt zum Problem, wenn zwar der Tab der CompleteSearch-Seite geschlossen wurde, die \$_SESSION aber weiterhin existiert, weil der Browser noch läuft; das lässt sich möglicherweise mit dem neuen URL-Parameter „reset“ erledigen)*
- *query_string != last_query_string*
- *Kann die Unterscheidung vom URL-Parameter „query“ abhängig gemacht werden (kommt er als URL-Parameter vor oder nicht?); dann dürfte er aber in allen Fällen != 2.1 nicht verwendet werden (geht das?)*

Grundlegendes Problem: alle übergebenen URL-Parameter müssen (durch relay.php?) eliminiert werden, da sonst History-Back nicht funktioniert (alternative Lösung wäre der sehnlich erwartete korrekte History-Mechanismus).

2.2 --

3 Suchanfrage

3.1 index.php?query=xxx

Berechne H_boxes[1] mit Treffer 1 bis \$AC->settings->hpp

Berechne W_boxes[1], F_boxes[i] mit Treffer 1 bis \$AC->settings->mcs

3.2 *processInput('xxx')*

Analog

4 Klick auf Vervollständigungen

4.1 *index.php?query=xxx*

Wie 3.1

H_boxes[1] mit Treffer 1 bis \$AC->settings->hpp

W_boxes[1], F_boxes[i] mit Treffer 1 bis \$AC->settings->mcs

4.2 *completion_link('xxx');*

Analog

5 Berechne zusätzliche Vervollständigungen in F_boxes[1] (more-Link)

Bsp.: Es werden 5 Vervollständigungen in F_boxes[1] angezeigt. Durch Klick auf „mehr“ sollen 15 Vervollständigungen angezeigt werden. Anders als bei „mehr“ in den Treffern werden alle 15 Vervollständigungen berechnet und die F_boxes[1] komplett neu gesetzt.

5.1 *index.php?qt=F&qi=1&mcs=15*

Berechne F_boxes[1] mit mcs=15

Die übrigen Boxen übernehme aus \$AC->result->H_boxes[1], W_boxes[1], F_boxes[2] ...

5.2 *more_link ('F', 1, 15)*

Berechne F_boxes[1] mit mcs=15 und setze F_boxes[1] entsprechend.

6 Blättern in Treffern

6.1 a) *index.php?qt=H&fh=15*

6.1 b) *index.php?qt=H&fh=15&bnm=R*

Berechne H_boxes[1] mit Treffer 15 bis 15 + hpp.

W_boxes[1], F_boxes[i] werden **nicht** berechnet, sondern aus \$AC->result->W_boxes[1], F_boxes[i] übernommen.

Bug: bisher wird der Titel falsch berechnet ('1 bis 15+hpp, Treffer sind aber korrekt), weil z. Z. nur bnm = R oder bnm = A unterstützt wird.

6.2 *navigation_link(15)*

Berechne H_boxes[1] mit Treffer 15 bis 15 + hpp

7 Berechne mehr Treffer (more-Link)

7.1 a) `index.php?qt=H&fh=1&hpp=30`

7.1 b) `index.php?qt=H&fh=16&hpp=15&bnm=A`

Berechne `H_boxes[1]` mit Treffer 1 bis 30

Übernehme `W_boxes[1]`, `F_boxes[i]` aus `$AC->result->W_boxes[1]`, `F_boxes[i]` ...

a) alle 30 Treffer werden neu berechnet

b) ein echtes „Append“ auf PHP-Seite (ähnlich dem JS-Mode)? D.h., nur die zusätzlichen Treffer werden berechnet und an die vorherigen Treffer aus `$_SESSION` anhängt.

7.2 `navigation_link(15, 'A', 1)`

Berechne `H_boxes[1]` mit Treffer 15 bis `15 + hpp`

`H_boxes[1][title] = 'Treffer 1 bis {14+hpp} ...'`

Hinweis: Andere Vorgehensweise als ohne JS, da `hpp` mit JS implizit gegeben ist

8 Wechseln der Sprache

8.1 `index.php?language=en`

Berechne alle Boxen neu (da sich die Überschriften ändern), aber mit den einzelnen `mcs` und `hpp` der letzten Anfrage

Hinweis: Das ist mit dem bisherigen Konzept nicht möglich; es werden die Werte aus `$AC->settings`

verwendet, nicht die der letzten Anfrage

8.2 *language_link('en')*

Analog

9 Rückkehr von options.php (genauer: change_options.php)

9.1 *index.php*

Berechne alle Boxen neu, entsprechend den neu gewählten Einstellungen (`$AC->settings`)

9.2 *index.php*

Analog

10 Browser-Navigation zurück / vor

10.1 *index.php?autocomplete_query=xxxy --> index.php?autocomplete_query=xxx*

Wie 1.1, mit query = vorherige Anfrage

H_boxes[1] mit Treffer 1 bis `$AC->settings->hpp`

W_boxes[1], F_boxes[i] mit Treffer 1 bis `$AC->settings->mcs`

10.2 *history_back()* / *history_forward()*

Analog

Überlegung: Besser wäre es, wenn nicht nur zum letzten Suchbegriff zurückgegangen würde, sondern auch zu den Zuständen der letzten Anfrage (also auch die mcs der einzelnen Boxen und das hpp der Treffer berücksichtigt würden)

Das scheint schon zu gehen; nochmal testen!

11 Rückkehr von externer Seite

Bei Rückkehr von einer externen Seite müssen die Fälle 3 bis 9 unterstützt werden

Überlegung: würde nicht am besten (wie mit JS) alles aus `$_SESSION` genommen?

Allerdings müßten dann die URL-Parameter ignoriert werden (die sind wiederum für den Modus ohne JS nötig)

Nachtrag: Müßte zum gleichen Ergebnis kommen, da die letzte Anfrage genau mit diesen URL-Parametern berechnet wurde

Beispiel:

11.1 *index.php?query=xxx&qt=F&qj=1&mcs=15*

Berechne `F_boxes[1]` mit `mcs=15`

Die übrigen Boxen übernehme aus `$AC->result->H_boxes[1]`, `W_boxes[1]`, `F_boxes[2]` ...

11.2 index.php

Übernahme H_boxes[1], W_boxes[1], F_boxes[i] aus \$AC->result->H_boxes[1], W_boxes[1], F_boxes[i]