

Multisites – flere sider på en database

Det har altid været muligt med PHP-Fusion at drive flere forskellige sider med basis i en database. Tricket består i at give sine tabeller forskelligt "fornavn". Dvs. at når man installerer sin side, så vælger man for eksempel fusion1 som fornavn til tabellerne til den ene side og fusion2 som fornavn til tabeller til den anden side.

Forudsætningen er naturligvis, at de forskellige sider ligger på den samme server – eller i det mindste bruger den samme databaseserver. Fornavnene til tabellerne angives i forbindelse med installationen i dette skærmbillede:



Trin 3: Database opsætning

Database adgangsoopsætning

Database værtsnavn:

Database brugernavn:

Database kodeord:

Databasenavn:

Tabellernes fornavn:

[Næste](#)

Men i PHP-Fusion er der desuden kommet en mulighed for at dele en eller flere tabeller imellem de to sider – for eksempel brugertabellen – og dermed også brugeroplysningerne. I dette afsnit beskriver vi, hvordan man gør det.

Nøglen til etablering af de såkaldte multisites skal findes i filen multisite_include.php der ligger i folderen **includes**. I denne fil defineres en lang række af standardtabeller ved globale variabelnavne, således at disse variabelnavne kan anvendes overalt, hvor PHP-Fusion kalder sin database.

Et udsnit af filen ser sådan her ud:

```
1 <?php
2 /*-----+
3 | PHP-Fusion Content Management System
4 | Copyright (C) 2002 - 2008 Nick Jones
5 | http://www.php-fusion.co.uk/
6 +-----+
7 | Filename: multisite_include.php
8 | Author: Nick Jones (Digitanium)
9 +-----+
10 | This program is released as free software under the
11 | Affero GPL license. You can redistribute it and/or
12 | modify it under the terms of this license which you
13 | can read by viewing the included agpl.txt or online
14 | at www.gnu.org/licenses/agpl.html. Removal of this
15 | copyright header is strictly prohibited without
16 | written permission from the original author(s).
17 +-----*/
18 // Cookie prefix
19 define("COOKIE_PREFIX", "fusion_");
20 // Database table definitions
21 define("DB_ADMIN", DB_PREFIX."admin");
22 define("DB_ARTICLE_CATS", DB_PREFIX."article_cats");
23 define("DB_ARTICLES", DB_PREFIX."articles");
24 define("DB_BBCODES", DB_PREFIX."bbcodes");
25 define("DB_BLACKLIST", DB_PREFIX."blacklist");
26 define("DB_CAPTCHA", DB_PREFIX."captcha");
27 define("DB_COMMENTS", DB_PREFIX."comments");
28 define("DB_CUSTOM_PAGES", DB_PREFIX."custom_pages");
29 define("DB_DOWNLOAD_CATS", DB_PREFIX."download_cats");
30 define("DB_DOWNLOADS", DB_PREFIX."downloads");
31 define("DB_FAQ_CATS", DB_PREFIX."faq_cats");
32 define("DB_FAQS", DB_PREFIX."faqs");
33 define("DB_FLOOD_CONTROL", DB_PREFIX."flood_control");
34 define("DB_FORUM_ATTACHMENTS", DB_PREFIX."forum_attachments");
35 define("DB_FORUM_POLL_OPTIONS", DB_PREFIX."forum_poll_options");
36 define("DB_FORUM_POLL_VOTERS", DB_PREFIX."forum_poll_voters");
37 define("DB_FORUM_POLLS", DB_PREFIX."forum_polls");
38 define("DB_FORUM_RANKS", DB_PREFIX."forum_ranks");
39 define("DB_FORUMS", DB_PREFIX."forums");
40 define("DB_INFUSIONS", DB_PREFIX."infusions");
41 define("DB_MESSAGES", DB_PREFIX."messages");
42 define("DB_MESSAGES_OPTIONS", DB_PREFIX."messages_options");
43 define("DB_NEW_USERS", DB_PREFIX."new_users");
44 define("DB_NEWS", DB_PREFIX."news");
```

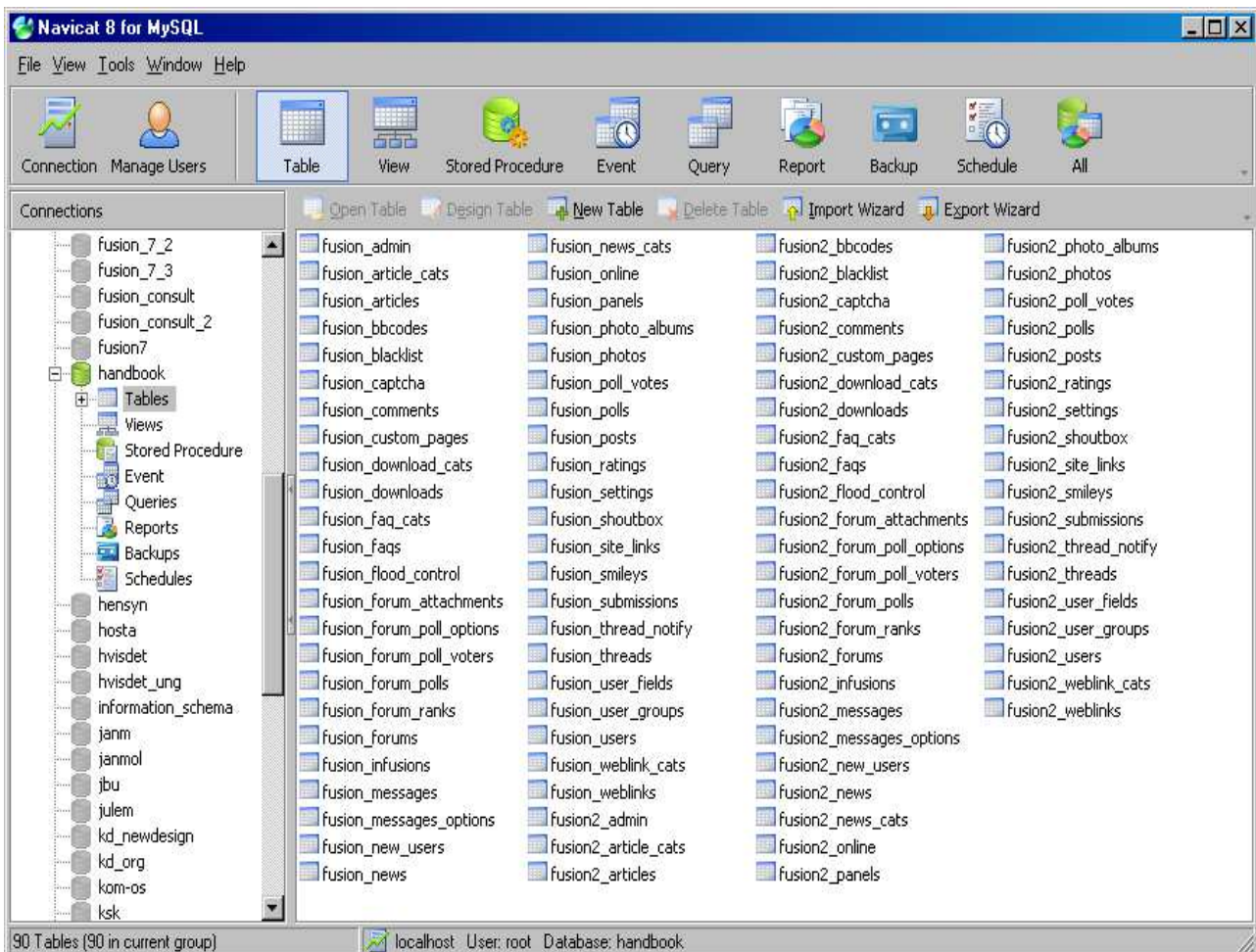
Som det fremgår, så defineres den globale variabel for tabellen **fusion_admin** her som **DB_ADMIN** og tilsvarende nedefter.

Når denne tabel så skal kaldes, anvendes variabelnavnet DB_ADMIN i stedet for det fulde tabelnavn. Det giver en sikkerhedsmæssig fordel, men det gør det altså også muligt at dele tabeloplysninger på tværs af sider installeret på samme database.

Det kunne for eksempel se sådan her ud (hvor vi kigger på tabellen fusion_downloads, der er defineret som **DB_DOWNLOADS**):

```
38 if ((isset($_GET['action']) && $_GET['action'] == "delete") && (isset($_GET['cat_id']) && isnum($_GET['cat_id']))) {
39     $result = dbquery("SELECT * FROM ".DB_DOWNLOADS." WHERE download_cat='".$_GET['cat_id']."'");
```

Hvis vi går ud fra, at vi har installeret vores testside her med standard-fornavnet fusion_ og en side mere på samme database med tabelnavnet fusion2_, så vil der under side nummer to dels være en række tabeller med fornavnet fusion, dels en række tabeller med fornavnet fusion2, som man kan se det på det næste skærmbillede.



I konfigurationsfilen (config.php) er det indskrevet, hvilken database, man benytter og hvilket fornavn tabellerne til denne side anvender. Og i filen **multisite_include.php** anvendes dette fornavn til at danne de globale variabelnavne.

Hvert site har sin egen **multisite_include.php**. Og det er den man skal have fat på, for at få de to sider til at dele information. I eksemplet her vil vi have de to sider til at dele brugeroplysninger.

På den første side ser config.php og de relevante linjer af multisite_include.php sådan her ud:

```

1 <?php
2 // database settings
3 $db_host = "localhost";
4 $db_user = "root";
5 $db_pass = "";
6 $db_name = "handbook";
7 $db_prefix = "fusion_";
8 define("DB_PREFIX", "fusion_");
9 ?>

```

```

60 define("DB_THREADS", DB_PREFIX."threads");
61 define("DB_USER_FIELDS", DB_PREFIX."user_fields");
62 define("DB_USER_GROUPS", DB_PREFIX."user_groups");
63 define("DB_USERS", DB_PREFIX."users");
64 define("DB_WEBLINK_CATS", DB_PREFIX."weblink_cats");
65 define("DB_WEBLINKS", DB_PREFIX."weblinks");
66 ?>

```

I config.php defineres variabelen \$db_prefix som fusion_ og anvendes i multisite_include.php via udtrykket DB_PREFIX. DB_PREFIX har med andre ord værdien fusion_. Hvis vi nu tager filen multisite_include.php fra den anden side – der er installeret på samme database men med en andet db_prefix, nemlig fusion2_, så ser de relevante linjer af den således ud:

```

60 define("DB_THREADS", DB_PREFIX."threads");
61 define("DB_USER_FIELDS", DB_PREFIX."user_fields");
62 define("DB_USER_GROUPS", DB_PREFIX."user_groups");
63 define("DB_USERS", DB_PREFIX."users");
64 define("DB_WEBLINK_CATS", DB_PREFIX."weblink_cats");
65 define("DB_WEBLINKS", DB_PREFIX."weblinks");
66 ?>

```

Den er med andre ord helt identisk med filen den første side. Men ændrer man nu de tre linjer, som har med brugere at gøre således:

```

60 define("DB_THREADS", DB_PREFIX."threads");
61 define("DB_USER_FIELDS", "fusion_user_fields");
62 define("DB_USER_GROUPS", "fusion_user_groups");
63 define("DB_USERS", "fusion_users");
64 define("DB_WEBLINK_CATS", DB_PREFIX."weblink_cats");
65 define("DB_WEBLINKS", DB_PREFIX."weblinks");
66 ?>

```

Så vil denne side og den oprindelige håndbogsside bruge de samme tabeller til at gemme brugeroplysninger i. Ser man før ændringen på brugeroplysninger fra den nye side, så ser de sådan her ud:

Og ser man på den efter ændringen, så ser den sådan her ud:

Brugere

Brugere :: Ikke aktiverede :: Udelukkede :: Tilføj ny bruger

Brugernavn	Brugerstatus	Valgmuligheder
Ny bruger	Bruger	Rediger - Udeluk - Slet
æøåtest	Bruger	Rediger - Udeluk - Slet

Vis alle A B C D E F G H I J K L M N O P Q R Vis alle
S T U V W X Y Z 0 1 2 3 4 5 6 7 8 9

Søg efter bruger:

De to sider deler nu brugeroplysninger.

Det samme kan man også gøre på andre områder som for eksempel nyheder eller artikler. Du skal bare huske på, at nogle tabeller forudsætter andre. Tabellen downloads kan ikke stå alene. Deler du **downloads**, skal du også dele **download_cats**. Tilsvarende med articles og news.