

УДК 004.415.3

В.В. Карбовський, доц. О.К. Каганюк, асистент П.С.Шолом
Луцький національний технічний університет

МУЛЬТИСАЙТИНГ В CMS DRUPAL

Карбовський В.В., Каганюк О.К., Шолом П.С. Мультисайтинг в CMS Drupal. В даній статті йдеться мова про налаштування мультисайтингу у системі керування вмістом Drupal. Наведені правила об'єднання таблиць бази даних MySQL.

Ключові слова: Мультисайтинг, система керування вмістом, CMS, Drupal, спільна база даних.

Карбовский Виктор Владимирович, Каганюк Алексей Казимирович, Шолом Павло Степанович. Мультисайтинг в CMS Drupal. В статье идет речь о настройке мультисайтинга в системе управления содержимым Drupal. Описанные правила объединения таблиц базы данных MySQL.

Ключевые слова: Мультисайтинг, система управления содержимым, CMS, Drupal, общая база данных.

Karbovskiy Viktor, Kaganjuk Oleksii, Sholom Pavlo. Multi-siting in CMS Drupal. The article is talking about setting multi-siting in content management system Drupal. Described association rules database tables MySQL.

Keywords: Multi-siting, content management system, CMS, Drupal, common database.

Постановка проблеми. Кількість користувачів Інтернет у світі та в Україні зокрема щороку збільшується в кілька разів. Сьогодні Всесвітня мережа стала невід'ємною частиною життя як у розвинених країнах, так і у таких, що розвиваються. Завдяки стрімким темпам зростання Інтернет, дедалі частіше у його користувачів постає питання щодо розроблення якщо не власного веб-сайту, то хоча б індивідуальної веб-сторінки. Саме через це ми все частіше чуємо слово «CMS».

Система керування вмісту веб-сайту (англ. Content Management System) – це програмне забезпечення для організації веб-сайтів чи в інших інформаційних ресурсів в Інтернеті чи окремих комп'ютерних мережах. CMS надають змогу швидко та ефективно розробляти веб-сайти, значно полегшують створення, наповнення та оновлення.

Drupal – безкоштовна модульна система з відкритим кодом, написана мовою програмування php. Розробником цієї системи є Дріс Байтаерт (Dries Buytaert) та спільнота розробників. Система Drupal може працювати у таких популярних операційних системах, як Windows, Mac OS X, Linux, власне, на будь-якій платформі, яка підтримує роботу веб-сервера Apache (починаючи з версії 1.3), або IIS (починаючи з версії IIS5+) та інтерпретатор мови php (починаючи з версії 4.7.5+), також потрібна наявність системи керування базами даних, наприклад, відкриті системи MySQL або PostgreSQL. Структура та потужна база модулів Drupal'у дає змогу порівняно швидко створювати потужні інтерактивні веб-сайти [1].

Аналіз останніх досліджень і публікацій. Необхідно зазначити, що аналізу проблем, які склалися в галузі веб-програмування, приділяється увага вітчизняним розробникам, серед яких Влад Савицький, Юрій Глушков, Олександр Мітасов та інші.

Невирішені частини проблеми. Кожен, хто використовує в своїх проектах CMS Drupal, рано чи пізно зустрічається з проблемою оновлення ядра та модулів при адмініструванні кількох сайтів, а також – економією місця на хостингу. Суть проблеми полягає в тому, що при потребі в оновленні важливих елементів, необхідно пророблювати однакові дії з кожним сайтом, що забирає цінний час і є не дуже практично.

Метою даної роботи є продемонструвати налаштування мультисайтингу для Drupal, що значно спростить адміністрування кількох сайтів, створених за допомогою вищезгаданої системи керування вмістом.

Основна частина. Мультисайтинг – одна з найбільш потужних особливостей Drupal, це можливість запускати декілька веб-сайтів, що використовують один і той же комплект файлів ядра системи [2](Рис.1.).

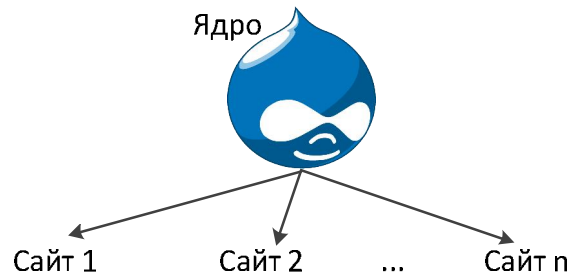


Рис. 1. Суть мультисайтингу

Для розв'язання проблеми перш за все необхідно зробити так, щоб новий сайт з доменом, наприклад, m.domain.ua, посилався на вже створений (domain.ua), тобто зробити його аліасом. Після цього обидва сайти будуть лежати в одній і ті ж директорії [3] (Рис.2).

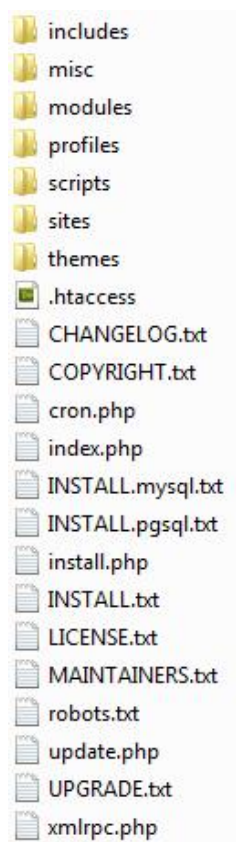


Рис. 2. Структура файлової системи Drupal

Призначення вкладених папок:

- files – тут містяться всі файли, завантажені відвідувачами на ваш сайт. Крім того, в даному каталозі можуть зберігатися персоналізовані файли налаштувань CMS Drupal та окремих модулів. Наприклад, змінені файли каскадних таблиць стилів, що використовуються темою оформлення, змінений JS-файл, файли. htaccess, robots. txt і ін.;
- includes – включає в себе файли з розширенням INC, що містять PHP-код ядра CMS Drupal;
- misc – додаткові файли, необхідні для коректного відображення вашого сайту: зображення, JS- та CSS-файли;
- modules – містить модулі, які входять до складу CMS Drupal. Ви можете помістити в цей каталог і нові модулі, проте правильніше для цього використовувати каталог sites \ all \ modules;

- profiles – включає в себе профілі установки CMS Drupal. Профіль установки – це текстовий файл з розширенням PROFILE, який визначає, які модулі будуть включені після інсталяції CMS Drupal і які налаштування будуть присвоєні цим модулям. Кожен файл профілю установки повинен знаходитися в окремій папці, розташованій в каталозі profiles;
- scripts – набір сценаріїв (файли з розширенням SH), призначених для віддаленого обслуговування CMS Drupal з допомогою SSH-підключення;
- sites – в даному каталозі містяться файли налаштувань CMS Drupal, а також модулі і теми оформлення, які ви додали після установки системи;
- themes – містить теми оформлення, які входять до складу CMS Drupal.

Основні файли, що знаходяться в кореневій папці:

- .htaccess – найчастіше містить відомості про те, до яких файлів (сценаріїв, сторінок HTML і т. д.) на вашому сайті зможуть отримати доступ відвідувачі.
- cron.php – даний сценарій виконує запуск планувальника завдань cron.
- index.php – головна сторінка CMS Drupal. Вона складається з 18 рядків коду і містить лише виклики функцій, які генерують і відображають основні регіони сторінки.
- robots.txt – включає в себе правила, на основі яких роботи пошукових систем повинні індексувати вміст вашого сайту.
- update.php – використовується для оновлення CMS Drupal.
- INSTALLmysql.txt – включає в себе опис процесу створення бази даних MySQL, яка буде використовуватися CMS Drupal після установки.

Всі операції проводяться в папці sites. Стандартно там знаходяться 2 папки: default (налаштування, модулі і теми для сайту за замовчуванням) та all (загальні модулі та теми).

Наступним етапом є створення в папці sites нової папки, ім'ям якої буде домен нового сайту (в даному випадку – m.domain.ua). Сюди необхідно скопіювати файл settings.php з папки default.

Необхідно зазначити, що у процесі пошуку файлів settings.php Drupal «обрізає» імена піддоменів зліва направо, а імена підпапок справа наліво. Перший знайдений файл буде використаний, решта проігноровано, тому дуже важливо вірно вказати ім'я папки для нового сайту.

Теми і модулі, що будуть використовуватись лише для нового сайту розміщуються у sites/m.domain.ua/themes та modules відповідно. Загальні теми і модулі - sites/all/themes та modules, а для вже створеного - sites/default/themes та modules.

Після цього необхідно відредагувати вищезгаданий файл settings.php.

- Якщо сайти будуть абсолютно незалежні, то в settings.php в \$db_url необхідно вказати окрему базу даних. Користувач також може бути будь-яким. Присутність або відсутність префікса ролі не грає.

- Якщо сайти використовувати спільну базу даних або таблиці.

○ Одна і та ж база даних (Рис.3.). У файлі settings.php в \$db_url вказуємо таку саму базу даних як на головному сайті, але обов'язково прописуємо інший префікс таблиць. Однак, з цим методом треба бути обережним, оскільки з кожним новим сайтом база даних буде значно збільшуватись. Наприклад, для головного сайту префікс s1_, а для нового – s2_. Для того, щоб новий сайт використовував спільну базу користувачів та їх профілів необхідно написати:

```
$db_prefix = array(
"default" => "s2_",
"users" => "s1_",
"sessions" => "s1_",
"authmap" => "s1_",
"profile_fields" => "s1_",
"profile_values" => "s1_",
);
```

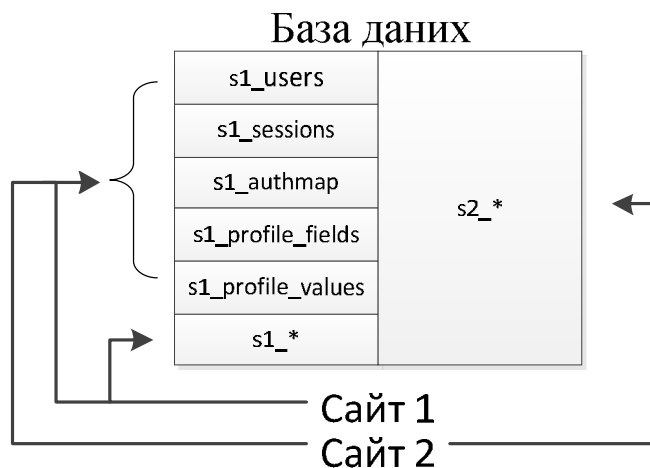


Рис. 3. Спільна база даних та деякі таблиці з неї

○ Різні бази даних, але деякі спільні таблиці (Рис.4.). Необхідно проробити все те саме, що і в попередньому підпункті, але ще й вказати назву бази даних, з якої беруться таблиці. Наприклад для отримання спільних користувачів та їх профілів на головному сайті (БД – s1, префікс – s1_) та новому (БД – s2, префікс – s2_), у файлі settings.php нового сайту прописується :

```
$db_prefix = array(  
"default" => "s2_",  
"users" => "s1.s1_",  
"sessions" => "s1.s1_",  
"authmap" => "s1.s1_",  
"profile_fields" => "s1.s1_",  
"profile_values" => "s1.s1_",  
);
```

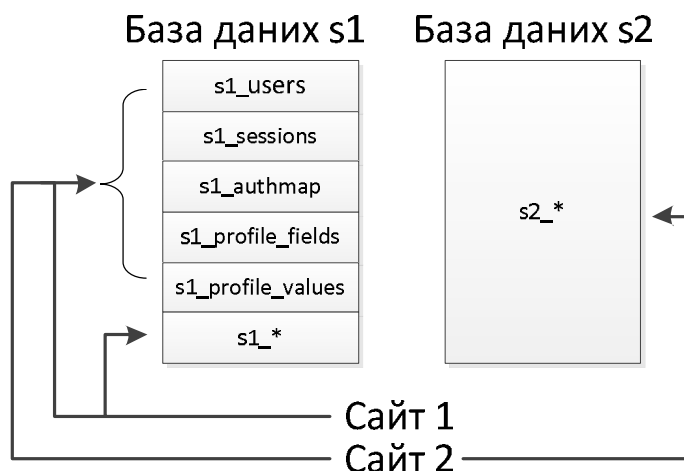


Рис. 4. Різні бази даних, але деякі спільні таблиці

При використанні спільних таблиць, необхідно дотримуватись певних правил, оскільки не всі таблиці можна робити спільними [4].

- Таблиці, які можна робити спільними:
 - filters та filter_formats – формати вводу;
 - locales_* – рядки перекладу;
 - role – ролі на сайті;

- permission – права ролей на хуки, що надаються модулями;
- users – користувачі сайту;
- users_roles – ролі користувачів сайту.
- Таблиці, які можна робити спільними при необхідності:
 - access – блокування по IP, імені користувача, e-mail;
 - authmap – таблиця для зовнішньої аунтетифікації (LDAP, OpenID);
 - flood – обмеження кількості запусків певних ділянок коду;
 - system – дані про встановлені модулі та теми;
 - sessions – сеанси користувачів.
- Таблиці, які можна робити спільними небезпечно:
 - comments – коментарі;
 - files та file_revisions – файли;
 - node та node_* – контент сайту;
 - term_* – терміни таксономії;
 - view_* – види;
 - vocabulary та vocabulary_node_types – словники таксономії.
- Таблиці, які можна робити спільними не можна:
 - blocks – таблиця блоків сайту;
 - blocks_roles – доступ ролей до блоків;
 - boxes – код блоків;
 - cache та cache_* – кеш;
 - history – останні непрочитані повідомлення;
 - menu – меню сайту;
 - sequences – внутрішній лічильник нод, коментарів, користувачів;
 - url_alias та url_alias_extra – аліаси, чисті посилання;
 - variable – змінні сайту;
 - watchdog – журнал помилок сайту;

Висновки. Основною задачею було налаштувати мультисайтинг в CMS Drupal. В результаті отримано два сайти, створені за допомогою даної системи керування вмістом, які використовують спільне ядро, деякі спільні модулі, а також таблиці бази даних MySQL для використання спільної бази користувачів та їх профілів. Обновлення буде відбуватись для обох сайтів разом, а також на хостингу зекономлено місце, в порівнянні з тим, якби не використовували мультисайтинг.

1. http://archive.nbuv.gov.ua/portal/chem_biol/nvnltnu/20_7/295_Kupar_20_7.pdf<http://musson.org.ua>
2. Рысецев М.В. Drupal 6.Bz drupal hand book, 2009-134с.
3. <http://camp09.drupal.ua/blog/multisiting>