

А.В. Горяинов, И.В. Пономарев

РАЗРАБОТКА СПЕЦИАЛИЗИРОВАННОГО МОДУЛЯ ДЛЯ CMS DRUPAL

Аннотация. Разработан специализированный модуль импорта данных на базе CMS Drupal. Стандартные возможности ядра Drupal расширяются дополнительным модулем, содержащим различные hook-функции. Описывается алгоритм действий, которые необходимы для разработки и создания собственного модуля импорта.

Ключевые слова: DRUPAL, система управления содержимым, CMS, модуль, hook, module, PHPExcel, функция, импорт, PHP, XLS.

Введение. Популярная система создания сайтов Drupal является модульной системой. Ее функциональные возможности реализуются с помощью подключаемых модулей. Большинство пользователей ограничивается использованием существующих официальных модулей или модулей, написанных сообществом Drupal. Но многие при создании сайта на базе системы управления содержимым (CMS) Drupal сталкиваются с проблемой, когда существующие модули не способны в полной мере обеспечить необходимую функциональность. Выход из этой ситуации – написание собственного модуля с использованием языка PHP, который обеспечит выполнение всех необходимых функций.

Постановка задачи. Необходимо проанализировать принципы создания модулей для CMS Drupal и разработать собственный модуль импорта данных.

Основная часть.

Правила организации модулей в Drupal.

Система управления сайтом Drupal построена по модульному принципу: компактный набор служебных функций (ядро) расширяется с помощью модулей. Модуль - это файл, содержащий набор функций на языке PHP. Код модуля может использовать функции, переменные и структуры ядра. В Drupal используется модель inversion of control (обратный вызов функции) - программа в нужное время вызывает модули,

обеспечивающие выполнение необходимых функций. Это обеспечивается обработчиками прерываний (hooks).

Новые возможности реализуются за счет установки дополнительных модулей, или благодаря разработке собственных модулей. Модули содержат hook-функции, вызываемые ядром при возникновении каких-либо событий.

Имя hook-функции должно состоять из двух частей: имени модуля и названия события. Каждый модуль для Drupal представляет собой два или более файла, которые располагаются в *sites/all/modules/назва_модуля*. Файлы с расширениями *.info* и *.module* - обязательные, *.install*-файлом можно пренебречь.

Файл с расширением *.info* должен содержать служебную информацию, без которой модуль не будет видно в системе. Также в файле *.info* располагаются три обязательных параметра: название модуля, его описание и версия ядра Drupal, и возможны необязательные параметры: минимальная версия PHP, зависимость от других модулей Drupal и др.

Файл *.install* предназначен для установки модуля на Drupal-сайт и удаления его из системы. В Drupal 7 *.install*-файл должен иметь функции *hook_install* и *hook_uninstall*, достаточно реализовать *hook_schema*, который определяет имя и структуру необходимых таблиц. Файл *.module* - важнейшая часть модуля содержит в себе исходный текст, именно в нем размещается большинство обработчиков прерываний.

Описание библиотеки PHPExcel.

Для написания модуля использовалась удобная объектно-ориентированная библиотека для работы с Excel-файлами *PHPExcel*. Библиотека *PHPExcel* имеет мощные средства для создания таблиц Excel и сохранения этих таблиц в разных форматах. Класс *PHPExcel* является ядром библиотеки. Он хранит в себе все листы (Sheet) книги Excel. Для установки активного листа (того с которым будем работать), нужно вызвать метод *setActiveSheetIndex()* и передать ему индекс листа по порядку. Метод *getActiveSheet()* вернет активный лист. Это объект класса *PHPExcel_Worksheet*. Класс *PHPExcel_Worksheet* имеет множество методов для работы с листами, ячейками, строками, столбцами и пр.

Для настройки стилей ячейки есть класс *PHPExcel_Style*. Добраться до него можно из активного листа (*\$aSheet*), вызвав метод *getStyle()*: *PHPExcel_Style getStyle ([string \$ pCellCoordinate = 'A1'])*

В качестве параметра передается строка с именем ячейки, для которой извлекается класс стилей. В классе стилей есть метод, который позволяет вынести все настройки в массив. Этот метод называется *applyFromArray()*. В качестве параметра ему передается массив с различными настройками.

Для чтения данных в библиотеке `RNRExcel` существует класс `RNRExcel_IOFactory`. В этом классе для загрузки есть статический метод *load()*; `RNRExcel PHP_Excel_IOFactory :: load (string $ pFilename)`.

Этот метод получает имя xls-файла, который нужно загрузить, а возвращает объект класса `RNRExcel`, с уже загруженными данными. После чего можно делать изменения в таблице с помощью вызова нужных методов класса `RNRExcel`.

Описание модуля и его работы

Разработанный модуль `sheetnode_phpexcel`, встроенный в пакет `sheetnode`, состоит из следующих файлов:

sheetnode_phpexcel.info – файл со служебной информацией;
sheetnode_phpexcel.module – основной файл созданного модуля;
sheetnode_phpexcel.import.inc – файл с реализацией вспомогательных функций.

В файле `sheetnode_phpexcel.info` содержится служебная информация. В основном файле модуля `sheetnode_phpexcel.module` реализована функция *hook_menu()*, в какой в цикле `foreach` в зависимости от значения *\$plugin* заполняется массив *\$items*. Значение *\$plugin* возвращает API-функция, которая предназначена для извлечения существующих `RNRExcel` плагинов *sheetnode_phpexcel_get_plugins()*.

Hook_menu() возвращает массив *\$items*, массив пунктов меню, которые добавляются модулем. Для импорта файла добавляются следующие адреса: *\$items ["node/add/\$format"]* - непосредственно при импорте файла; *items['admin/settings/sheetnode/phpexcel']* - ссылка на редактирование настроек модуля импорта файла, расположенного в профиле администратора Drupal, в подразделе «Настройка сайта».

В массив *\$items* добавляется массив, содержащий настройки данных пунктов меню. Ключом массива является ссылки пунктов меню.

При выборе одного из пунктов меню вызывается функция, которая указана в элементе *page_callback* вложенного массива. Это функция *drupal_get_form()*, которая получает в качестве аргументов данные записанные в элементе *page arguments*. В результате Drupal вернет следующие

формы: *_sheetnode_phpexcel_import_form* и формат *\$format* при импорте файла; *_sheetnode_phpexcel_settings* при настройке модуля.

Функция *batch_set(\$batch)* устанавливает и передает операции, заданные в пакете *\$batch*, которые должна выполнить функция *batch_process(\$destination)*.

Функция *_sheetnode_phpexcel_batch_import_file()*, которая реализована в файле *sheetnode_phpexcel.import.inc*, в качестве параметра принимает *\$callback* и ссылки на контекст *&\$context* (в котором хранится считываемая информация, соответствующая текущему состоянию). Если книга еще не загружена (*empty(\$context['sandbox']['workbook']) = TRUE*), то эта функция загружает новую книгу *workbook* и получает количество писем - *worksheets*. Если книга загружена, то продолжаем с ней работать - создается *sheetnode* для текущего листа. Затем вызывается функция, переданная через параметр *\$callback* (если функции *_sheetnode_phpexcel_batch_import_file()* был передан параметр *\$callback*) и сохраняется *sheetnode*. *\$callback()* позволяет другим модулям изменить *sheetnode*, или сделать другие дополнительные действия

```
if (! empty ($ callback) && function_exists ($ callback))
    { $ callback ($ node, $ params, $ context) }
```

Далее обновляется информация прогресса. При создании *sheetnode* вызывается функция *_sheetnode_phpexcel_import_do()*, которая «читает» excel-файл и заполняет SocialCalc структуру *\$sc*. При заполнении структуры вызывается функция *_sheetnode_phpexcel_import_cell()*, которая проходит по всем ячейкам, определяет тип содержимого и заполняет *\$sc['cells'][\$cell>getCoordinate()]* с помощью дополнительного массива *\$c*.

Подключение и пример работы модуля Sheetnode РНРExcel

Созданные файлы модуля необходимо скопировать в папку */modules/sheetnode/modules/sheetnode_phpexcel/*. Модуль импорта/экспорта Sheetnode РНРExcel включается на странице «Администрирование → Конструкция сайта → Модули». Далее, необходимо перейти на страницу создания материала и выбрать новый пункт «Импорт из XLS файла. Создание нового письма с содержанием существующих таблиц». Выбирается *.xls файл, например *ant.xls*, содержание которого приведено на рисунке 1.

| № | А | В | С | D | E |
|---|---|---------|---|--------------------|-------|
| 1 | № | Артикул | Название | Картинки | Цена |
| 2 | 1 | 201 | Кухня угловая - венге 3400мм x 850мм | /img/catalog/1.jpg | 87000 |
| 3 | 2 | 202 | Кухня крем-бордо 3900мм x 600мм | /img/catalog/2.jpg | 6500 |
| 4 | | 203 | Кухня угловая с барной стойкой | /img/catalog/3.jpg | 2345 |

Рисунок 1 - Содержимое файла для импорта

После нажатия кнопки «Отправить» можно увидеть процесс загрузки файла (или пакета файлов) в виде графической шкалы. Далее высветится сообщение об успешном выполнении операции или сообщение с текстом ошибки. В данном случае файл импортировано успешно. Вид страницы с импортированным файлом представлен на рисунке 2.

Лист 1

| № | Артикул | Название | Картинки | Цена |
|---|---------|---|--|-------|
| 1 | 201 | Кухня угловая - венге 3400мм x 850мм |  | 87000 |
| 2 | 202 | Кухня крем-бордо 3900мм x 600мм |  | 6500 |
| | 203 | Кухня угловая с барной стойкой |  | 2345 |

Рисунок 2 - Вид страницы с импортированным файлом

Пример использования разработанного модуля можно увидеть на сайте ant.dr.ua.

Выводы. Рассмотрены основные особенности внутренней организации модульной структуры Drupal, принцип действия hook-функций. Стандартные возможности ядра расширяются разработанным дополнительным модулем, с использованием специализированной библиотеки. Описана разработка модуля и его работа.

ЛИТЕРАТУРА

1. Тодд Томлинсон. CMS Drupal 7. Руководство по разработке системы управления веб-сайтом. Пер. с англ. - М.: Издательский дом «Вильямс», 2011 – 560 с.
2. Victor Kane. Leveraging Drupal: getting your site done right. – Wrox, 2009 – 480p.