Выбор и настройка модулей
для конструирования динамических веб-сайтов

Drupal

Создание и управление сайтом

O'REILLY®

Анжела Байрон, Эддисон Берри,
Натан Хог, Джефф Итон,
Джеймс Уокер, Джефф Роббинс

По договору между издательством «Символ-Плюс» и Интернет-магазином «Books.Ru – Книги России» единственный легальный способ получения данного файла с книгой ISBN 978-5-93286-174-5, название «Drupal: создание и управление сайтом» – покупка в Интернет-магазине «Books.Ru – Книги России». Если Вы получили данный файл каким-либо другим образом, Вы нарушили международное законодательство и законодательство Российской Федерации об охране авторского права. Вам необходимо удалить данный файл, атакже сообщить издательству «Символ-Плюс» (piracy@symbol.ru), где именно Вы получили данный файл.

Using Drupal

Angela Byron, Addison Berry,
Nathan Haug, Jeff Eaton,
James Walker, Jeff Robbins

O'REILLY®

Drupal

Создание и управление сайтом

Анжела Байрон, Эддисон Берри,
Натан Хог, Джефф Итон,
Джеймс Уокер, Джефф Роббинс

СИМВОЛ®

Санкт-Петербург — Москва
2010

Анжела Байрон, Эддисон Берри, Натан Хог,
Джефф Итон, Джеймс Уокер, Джефф Роббинс

Drupal: создание и управление сайтом

Перевод А. Киселева

Главный редактор	*А. Галунов*
Зав. редакцией	*Н. Макарова*
Выпускающий редактор	*П. Щеголев*
Редактор	*Ю. Бочина*
Корректор	*С. Минин*
Верстка	*К. Чубаров*

Байрон А., Берри Э., Хог Н., Итон Дж., Уокер Дж., Роббинс Дж.

Drupal: создание и управление сайтом. – Пер. с англ. – СПб.: Символ-Плюс, 2010. – 576 с., ил.

ISBN 978-5-93286-174-5

Drupal – это открытая система, которая используется для создания веб-сайта, а также платформа управления его содержимым с помощью дополнительных модулей расширения. Ценность Drupal заключается в том, что он позволяет разработчикам легко и быстро комбинировать модули, создавая эффективные и богатые возможностями сетевые ресурсы, практически не написав ни строчки программного кода.

С помощью этой книги читатель сможет не только составить общее представление о системе Drupal, ключевых модулях и их настройке, но и разобраться, как среди всего их многообразия выбрать тот, который оптимально соответствует стоящей перед разработчиком задаче. В книге рассматривается создание сайтов следующих типов: каталог вакансий, wiki-проект, фотогалерея, интернет-магазин, сайт с коллективным информационным наполнением и поддержкой мультиязычных версий. Рассказ о каждом проекте сопровождается пошаговыми инструкциями и подробным описанием действий.

Как начинающий пользователь, так и профессионал смогут почерпнуть из книги массу полезной информации и с помощью Drupal приступить к созданию сайта своей мечты.

ISBN 978-5-93286-174-5
ISBN 978-0-596-51580-5 (англ)

© Издательство Символ-Плюс, 2010
Authorized translation of the English edition © 2009 O'Reilly Media Inc. This translation is published and sold by permission of O'Reilly Media Inc., the owner of all rights to publish and sell the same.

Все права на данное издание защищены Законодательством РФ, включая право на полное или частичное воспроизведение в любой форме. Все товарные знаки или зарегистрированные товарные знаки, упоминаемые в настоящем издании, являются собственностью соответствующих фирм.

Издательство «Символ-Плюс». 199034, Санкт-Петербург, 16 линия, 7,
тел. (812) 380-5007, www.symbol.ru. Лицензия ЛП N 000054 от 25.12.98.
Налоговая льгота – общероссийский классификатор продукции
ОК 005-93, том 2; 953000 – книги и брошюры.

Подписано в печать 28.12.2009. Формат 70×100 1/16. Печать офсетная.
Объем 36 печ. л. Тираж 1200 экз. Заказ №
Отпечатано с готовых диапозитивов в ГУП «Типография «Наука»
199034, Санкт-Петербург, 9 линия, 12.

Оглавление

Об авторах 13

Предисловие 16

Введение 18

Глава 1. Обзор системы Drupal 29

Что такое Drupal? 29

Кто ею пользуется? 30

Возможности, которые поддерживает Drupal 32

Краткая история систем управления содержимым 33

Исторический экскурс в процесс создания веб-сайта 33

Эпоха сценариев и баз данных 35

Революция в управлении содержимым 36

Как работает Drupal 38

Модули 39

Пользователи 40

Содержимое (узлы) 41

Способы организации содержимого 43

Типы содержимого поддержки 45

Получение справки 46

В заключение 48

Глава 2. Запускаем Drupal 49

Задача 51

Примечания к реализации 51

Основы 51

Простота редактирования и обработки изображений 52

В центре внимания: управление содержимым 53

Содержимое 53

Комментарии 57

Навигация 58

Блоки 59

Практика: управление содержимым 62

Создание содержимого 63

Управление навигацией по сайту 66

Настройка блоков 67
В центре внимания: модули 71
Страница администрирования модулей 71
Поиск и установка модулей 72
Удаление модулей 75
Практика: работа с модулями 75
Модуль Path 75
Модуль Administration Menu 77
В центре внимания: права доступа 80
Настройка прав доступа пользователей 81
Профили пользователей 84
Настройки пользователей 85
Блокирование пользователей 85
Практика: создание ролей 86
Практика: настройка разрешений 88
Практика: форма с контактной информацией 96
В центре внимания: модуль Taxonomy 99
Практика: блог 101
В центре внимания: инструменты модерирования содержимого 108
Инструменты автоматизированного определения спама 108
Инструменты модерирования содержимого вручную 110
В центре внимания: темы оформления 111
Поиск тем оформления 111
Установка тем оформления 111
Настройка тем оформления 112
Блоки и области 116
Настройка темы оформления административного раздела 116
Практика: придание сайту неповторимого внешнего вида 117
В центре внимания: редактирование содержимого и работа с изображениями 118
Редактирование содержимого 119
Работа с изображениями 120
В центре внимания: форматы ввода и фильтры 122
Практика: настройка модуля FCKeditor 127
В заключение 135

Глава 3. Каталог вакансий 136

Задача 137
Примечания по реализации 138
Нестандартные формы ввода 138
Отображение содержимого 138
Выгрузка файлов 138

В центре внимания: CCK 139
Поля 142
Виджеты 143
Форматтеры 144
Практика: CCK 145
Практика: тип содержимого Job (вакансия) 146
Практика: настройка отображения поля 154
Практика: тип содержимого Job Application (заявка) 158
В центре внимания: модуль Views 164
Отображения 166
Элементы представления 169
Практика: модуль Views 172
Представление Jobs (вакансия) 174
Представление Job Applications (заявки) 188
Дополнительно 204
В заключение 205

Глава 4. Обзоры продуктов 207

Задача 208
Примечания к реализации 209
Информация о продукте 209
Оценка продуктов 210
Реализация поиска 211
Первый шаг: основные описания продуктов 211
Создание типа содержимого Product Review 212
В центре внимания: модуль Amazon 216
Что включено? 216
Региональные настройки 217
Настройки ссылочной информации 218
Практика: добавление поля Amazon 218
Добавление поля Product ID 219
Поиск идентификаторов продуктов 221
В центре внимания: Voting API и Fivestar 222
Практика: добавление оценок 224
Добавление поля Rating 224
Включение оценок посетителей 225
Практика: создание представления со списком продуктов 228
В центре внимания: модуль Search 233
Важность выполнения регулярных процедур 235
Поиск с помощью модуля Views 236
Практика: добавление возможности поиска в список продуктов 236
В центре внимания: модуль CSS Injector 239

Практика: улучшение внешнего вида ... 240
Настройка параметров отображения полей CCK ... 241
Настройка модуля CSS Injector ... 241
Дополнительно ... 243
В заключение ... 244

Глава 5. Wiki ... 245

Задача ... 247
Примечания к реализации ... 247
Функциональные возможности Wiki ... 247
Простота форматирования текста ... 248
Простота создания ссылок ... 249
Отслеживание изменений ... 249
Удобочитаемые адреса URL ... 250
Список изменений ... 250
Практика: первые шаги ... 250
Создание типа содержимого Wiki ... 250
Удаление информации об авторе при отображении ... 252
Настройка прав доступа ... 252
Практика: формат ввода wiki ... 253
Настройка фильтров ... 254
Создание формата ввода wiki ... 256
Настройка разрешений формата ... 257
Добавление содержимого ... 258
В центре внимания: модуль Pathauto ... 261
Практика: модуль Pathauto ... 264
Настройка параметров ... 265
В центре внимания: отслеживание редакций в Drupal ... 267
Практика: редакции и модуль Diff ... 268
Включение механизма редакций по умолчанию ... 268
Настройка разрешений ... 269
Просмотр редакций и переключение между ними ... 269
Использование модуля Diff ... 270
Практика: создание представлений New pages и Recent edits ... 272
Список Recent posts ... 273
Список Recent edits ... 274
Дополнительно ... 277
В заключение ... 278

Глава 6. Управление процессом публикации ... 280

Задача ... 281
Примечания к реализации ... 282
Процесс публикации ... 282

Извещение по электронной почте ... 282
Обзорные страницы ... 283
Практика: первые шаги ... 283
В центре внимания: действия и триггеры ... 285
Практика: действия и триггеры ... 287
Настройка действий ... 287
Назначение триггеров ... 290
В центре внимания: модуль Workflow ... 291
Практика: определение процесса публикации ... 294
В центре внимания: модуль Workspace ... 299
Практика: создание рабочих пространств ... 300
В центре внимания: модуль Views Bulk Operations ... 301
Практика: создание страницы администрирования ... 303
Создание действий процесса публикации ... 303
Настройка представления ... 304
Дополнительно ... 305
В заключение ... 307

Глава 7. Фотогалерея ... 308

Задача ... 309
Примечания к реализации ... 309
Выгрузка фотографий ... 310
Создание миниатюр ... 310
Фотогалереи ... 310
В центре внимания: модуль ImageField ... 310
Настройка ... 311
Практика: выгрузка фотографий ... 315
Тип содержимого Photo ... 315
Галереи изображений ... 318
В центре внимания: модуль ImageCache ... 321
Наборы правил и операции ... 323
Использование наборов правил ... 327
Разрешение проблем с модулем ImageCache ... 329
Практика: модуль ImageCache ... 331
Создание наборов правил для ImageCache ... 332
Настройка отображения поля Photo (Фотография) ... 334
Улучшение качества изображений ... 335
Практика: представление gallery ... 336
Практика: блок с последними фотографиями ... 342
Практика: модуль Custom Pagers ... 346
Дополнительно ... 348
В заключение ... 350

Глава 8. Многоязычные сайты ... 351
Задача ... 353
Примечания к реализации ... 353
Форумы ... 353
База знаний ... 354
Перевод пользовательского интерфейса ... 354
Перевод содержимого, созданного пользователями ... 354
В центре внимания: средства интернационализации в ядре ... 355
Модуль Locale ... 355
Перевод содержимого ... 360
Практика: установка перевода ... 361
Практика: настройка модуля Locale ... 366
Настройка механизма выбора языка ... 366
Переключатель языка ... 367
Практика: переводимое содержимое ... 368
В центре внимания: модуль Localization Client ... 369
Практика: перевод интерфейса ... 370
Использование модуля Locale ... 371
Использование модуля Localization Client ... 372
Практика: перевод содержимого ... 376
Перевод ... 376
В центре внимания: Internationalization ... 379
Выбор содержимого ... 380
Модуль Strings ... 383
Переменные сайта, зависящие от языка ... 383
Вспомогательные модули ... 383
Синхронизация ... 384
Практика: особенности пакета Internationalization ... 385
Выбор содержимого ... 385
Переменные сайта ... 386
Типы содержимого ... 389
Таксономия ... 393
Дополнительно ... 395
В заключение ... 397

Глава 9. Управление событиями ... 399
Задача ... 400
Примечания к реализации ... 401
Управление событиями ... 401
Слежение за численностью участников ... 402
Практика: первые шаги ... 402
Создание типа содержимого Event ... 403
Права доступа ... 404

В центре внимания: модуль Date 405
Модуль Date Timezone 406
Типы полей для хранения дат 406
Виджеты ввода дат 407
Настройки модуля Date 408
Практика: добавление дат 414
Настройка модуля Date 414
Добавление поля даты 414
Практика: представление для отображения предстоящих событий 417
В центре внимания: модуль Calendar 421
Тип представления Calendar 422
Практика: представление Calendar 422
В центре внимания: модуль Flag 426
Практика: настройка модуля Flag 427
Практика: представление Attendee 429
Дополнительно 431
В заключение 432

Глава 10. Интернет-магазин 434

Задача 435
Примечания к реализации 436
В центре внимания: пакет Ubercart 437
Ubercart – core 437
Ubercart – core (optional) 439
Ubercart – extra 444
В центре внимания: панель администрирования Ubercart 445
Практика: настройка магазина 448
Начальная настройка 448
В центре внимания: модули Products, Product Classes и Attributes 450
Практика: создание продуктов 452
Начальная настройка 452
Настройка классов продуктов 454
Настройка атрибутов продуктов 455
Настройка параметров продуктов 458
Настройка каталога 459
В центре внимания: процесс оформления заказа 464
Практика: обработка заказов 466
Покупательская корзина 467
Налоги 469
Доставка 469
Оплата 470

Размещение тестового заказа ... 471
Исполнение заказа ... 474
Управление доступом ... 477
Дополнительно ... 479
В заключение ... 480

Глава 11. Оформление сайта ... 482

В центре внимания: механизм поддержки тем оформления ... 483
Файлы .info ... 486
Области ... 487
Свойства ... 488
CSS ... 489
JavaScript ... 491
Файлы шаблонов ... 491
Файл template.php ... 493
Практика: создание собственной темы ... 496
Создание копии темы ... 497
Изменение CSS ... 500
В центре внимания: файлы шаблонов ... 502
Практика: работа с файлами шаблонов ... 503
Модификация файла шаблона ... 503
Оформление отдельных типов содержимого ... 507
Переопределение файла шаблона модуля ... 508
Создание новой области ... 511
В центре внимания: дополнительные возможности переопределения ... 513
Переменные шаблона ... 513
Функции темы оформления ... 514
Практика: использование файла template.php для переопределения ... 516
Переопределение переменной шаблона ... 516
Переопределение функции темы оформления ... 517
Дополнительно ... 520
В заключение ... 521
Ссылки ... 522

Приложение A. Установка и обновление Drupal ... 524

Приложение B. Выбор модулей ... 546

Приложение C. Модули и темы оформления, использованные в этой книге ... 563

Алфавитный указатель ... 567

Об авторах

Анжела Байрон (Angela Byron) – убежденная сторонница движения за распространение программного обеспечения с открытыми исходными текстами; она живет и дышит проектом Drupal. Для нее все началось в студенческом лагере Google Summer of Code в 2005 году, и с тех пор она полностью отдала себя работе в сообществе Drupal. В ее обязанности входит разработка нового программного кода и оценка исправлений, создание и передача в проект новых модулей и тем оформления, тестирование и проверка программного кода, работа над документацией и поддержка пользователей в форумах и каналах IRC.

Анжела входит в состав совета директоров ассоциации Drupal и является членом группы сопровождения Drupal 7. Она способствует дальнейшему росту сообщества, выдвигая инициативы по привлечению новых добровольцев, например представляет проект Drupal в программах Google Summer of Code и Google's Highly Open Participation (GHOP). Анжела пользуется популярностью как лектор по многим темам, особенно это относится к теме «Женщины в движении за открытые исходные тексты». На сайте Drupal.org Анжела известна под псевдонимом «webchick».

Эддисон Берри (Addison Berry) принимает участие во многих направлениях развития программного обеспечения Drupal и сообщества. Она вносит исправления в ядро Drupal, поддерживает несколько модулей и активно участвует в нескольких программах, таких как группа Drupal Dojo и программа GHOP.

В проекте Drupal Эддисон является руководителем команды документирования, помогает поддерживать сайт Drupal.org и является постоянным членом генеральной ассамблеи ассоциации Drupal. Она создала множество учебных печатных и видеоруководств, охватывающих все аспекты Drupal – от участия в жизни сообщества до создания программного кода. На сайте Drupal.org Эдди известна под псевдонимом «add1sun».

Натан Хог (Nathan Haug) – один из ведущих разработчиков пользовательского интерфейса в проекте Drupal. Интерес к дизайну интерфейса и разработке программного обеспечения привел его к получению степени бакалавра по специальностям «визуальные взаимодействия» и «информатика». Он внес существенные улучшения в пользовательский интерфейс версии Drupal 6, включая реализацию механизма буксиров-

ки (drag-and-drop) в Drupal и фреймворк для поддержки простых Ajax-подобных операций.

В проекте Drupal Натан считается ведущим разработчиком сценариев JavaScript. В 2007 году он возглавил группу разработки компании SonyBMG, занимавшейся созданием семейства веб-сайтов на платформе Drupal для всех музыкантов, сотрудничающих с SonyBMG. Большую часть своего времени он тратит на работу над популярными модулями, такими как Fivestar и Webform, или над улучшениями в ядре Drupal. На сайте Drupal.org Нат известен под псевдонимом «quicksketch».

Джефф Итон (Jeff Eaton) более десяти лет занимается разработкой обычных и веб-приложений. Принимал участие в самых разных проектах, от веб-порталов для сообществ и некоммерческих организаций до клиент-серверных приложений для предприятий розничной торговли и масштабируемых веб-приложений для таких компаний, как Dow AgroSciences и Prudential Real Estate.

В 2005 году он начал заниматься разработкой решений на основе открытой платформы управления содержимым Drupal. За прошедшие с тех пор годы он стал в проекте Drupal одним из основных разработчиков, специализирующимся на разработке архитектуры и прикладного программного интерфейса. Занимая должность консультанта в компании Lullabot Consulting (LLC), он помогал проектировать и создавать программную инфраструктуру сайтов на основе Drupal, включая музыкальный портал MTV UK, платформу сайтов для музыкантов SonyBMG и сайт инновационной компании Fast Company. На сайте Drupal.org Джефф известен под псевдонимом «eaton».

Джеймс Уокер (James Walker) заведует отделом образования в компании Lullabot – курирует общественные симпозиумы и семинары, проводимые компанией, и частное обучение системе Drupal, соединяя в себе страсть к решению технических проблем и обучению. Будучи лидером сообщества Drupal, Джеймс является одним из основных членов некоммерческой ассоциации Drupal и группы безопасности Drupal. Как давний член сообщества Drupal Джеймс занимается поддержкой более чем десятка модулей и внес бесчисленное множество исправлений в ядро Drupal.

Будучи давним сторонником идеи открытых исходных текстов и открытых стандартов, Джеймс уже длительное время координирует деятельность проекта Drupal с деятельностью других сообществ, таких как Jabber/XMPP и появившийся совсем недавно OpenID. Как талантливого лектора Джеймса часто приглашают в качестве докладчика на самые разнообразные технические конференции. Его содержательные и наполненные юмором лекции являются самыми посещаемыми на конференциях DrupalCons, начиная с самой первой из них, которая проводилась четыре года тому назад. На сайте Drupal.org Джеймс известен под псевдонимом «walkah».

Джефф Роббинс (Jeff Robbins) является сооснователем и президентом компании Lullabot. В 1993 году Джефф основал одну из самых первых компаний по разработке веб-приложений и разрабатывал сайты для Ринго Старра (Ringo Starr), MTV и Нью-Йоркского музея современного искусства. Кроме того, большую часть 1990-х годов Джефф занимался продвижением музыкального коллектива Orbit, участвовал в музыкальном фестивале Lollapalooza и сочинил рок-композицию, вошедшую в лучшую десятку.

Джефф отвечает за поддержку популярного подкаста Drupal компании Lullabot и снискал заслуженную известность в сообществе. Кроме того, он создал множество широко используемых модулей и тем оформления Drupal, включая модули ConTemplate, LoginToboggan, тему Zen и инструмент Theme Developer. На сайте Drupal.org Джефф известен под псевдонимом «jjeff».

Предисловие

Модульная архитектура и открытость Drupal – прикладной платформы на языке PHP и системы управления содержимым – обеспечивают ей высокую популярность среди сотен и тысяч веб-разработчиков по всему миру. Более 900 человек способствовали выходу версии Drupal 6, поставляя программный код и идеи, еще больше разработчиков занимаются разработкой и сопровождением более 2000 модулей, которые могут использоваться для расширения функциональных возможностей этой системы.

Численность и активность сообщества пользователей и разработчиков Drupal в комбинации с ее мощью как платформы позволяют творить чудеса. Каждый день появляются новые и усовершенствуются существующие модули. Эти модули создаются, чтобы соответствовать последним тенденциям развития Интернета, и реализуют совершенно новые парадигмы – одним словом, проект Drupal продолжает развиваться в самых разных направлениях.

Ценность всего этого множества модулей заключается в том, что они позволяют разработчикам легко и быстро создавать эффективные и богатые возможностями веб-сайты, не становясь при этом программистами. Миллионы людей используют Drupal для создания личных блогов, корпоративных веб-сайтов, фотогалерей, справочных ресурсов, сайтов для размещения объявлений о поиске работы, веб-конференций и многих других.

К сожалению, основную сложность для многих администраторов сайтов и даже для опытных разработчиков системы Drupal представляет невозможность опробовать и понять назначение всего набора модулей и постоянно расширяющихся возможностей Drupal. Какие модули необходимы для построения новостного веб-сайта? Какие модули следует использовать для построения сайта в локальной сети? Какими модулями лучше не пользоваться, потому что имеются более удачные варианты? Какие модули можно использовать при создании действительно крупных веб-сайтов, обслуживающих миллионы запросов в день? Путешествие по миру Drupal может оказаться делом весьма нелегким.

Эта книга сэкономит время, необходимое на начальном этапе для вхождения в курс дела, и поможет сразу же погрузиться в удивительный мир Drupal. Она послужит вам прекрасным помощником, подсказыва-

ющим, как быстро создавать различные веб-сайты с использованием некоторых наиболее распространенных модулей Drupal. Независимо от того, являетесь ли вы опытным программистом или начинающим веб-разработчиком, эта книга содержит массу полезной для вас информации. Я обещаю, что к концу чтения этой книги вы будете гораздо лучше подготовлены для постройки сайта вашей мечты. И все это благодаря Drupal.

Дриз Байтаерт (Dries Buytaert),
основатель и руководитель проекта Drupal,
июль 2008

Введение

Для кого предназначена эта книга?

- Если вы ведущий разработчик и никогда раньше не слышали это странное слово «Drupal», но хотите понять, что оно означает, эта книга для вас.
- Если ваш босс сказал вам: «Нам необходимо создать быстродействующий сайт X!», где «X» – это фотогалерея, каталог товаров, сайт электронной коммерции или любой другой проект из числа тех, что рассматриваются в этой книге, эта книга для вас.
- Если вы уже имели дело с Drupal, но испытываете трудности при выборе из огромного многообразия модулей и нуждаетесь, чтобы кто-то подсказал, на какие из них стоит обратить внимание, тогда эта книга для вас.
- Если вы считаете себя знатоком системы Drupal, но стремитесь больше узнать о малоизвестных модулях и познакомиться с передовыми приемами создания мощных веб-сайтов на основе Drupal, эта книга для вас.

Если вы совершенно ничего не знаете о том, как создаются веб-сайты и устанавливаются сценарии веб-приложений, то эта книга, вероятно, *не* для вас. Мы предполагаем, что в ваш рабочий лексикон входят такие незамысловатые аббревиатуры, как PHP, FTP, URL, ZIP и HTML. Эта книга *не* для вас, если вы интересуетесь внутренним устройством ядра системы и особенностями работы функций прикладного программного интерфейса Drupal, поскольку основное внимание в этой книге уделяется не созданию новых модулей, а комбинированию *существующих* с целью построения функциональных веб-сайтов.

Если вы не относите себя ни к новичкам, ни к профессионалам, мы рассчитываем, что эта книга послужит ценным руководством по созданию работоспособных веб-сайтов на основе Drupal.

Исходные требования для работы с книгой

Вам необходимо иметь доступ к компьютеру или серверу, где установлены и настроены PHP, веб-сервер (предпочтительнее Apache) и база дан-

ных (рекомендуется MySQL). Для разработки на локальном компьютере существует несколько пакетов программного обеспечения, включающих необходимые Apache/MySQL/PHP, такие как WAMP для Windows (*http://www.wampserver.com*) или MAMP для Mac (*http://www.mamp.info*).

В списке *http://drupal.org/hosting* приводится перечень компаний, предоставляющих услуги веб-хостинга и обеспечивающих поддержку Drupal, а на странице *http://drupal.org/requirements* подробно описываются требования системы Drupal.

Кроме того, вам потребуется установить систему Drupal, а в практических разделах глав предполагается, что вы используете исходный программный код, распространяемый вместе с книгой. В приложении A даются некоторые базовые инструкции, но если вам придется столкнуться с какими-либо проблемами или потребуются более подробные инструкции, загляните в руководство по установке Drupal 6 по адресу *http://drupal.org/getting-started/install*. Если вы не используете исходный программный код, поставляемый вместе с книгой, в приложении C вы найдете полный список всех модулей и тем, использовавшихся в каждой главе, что позволит создать код самостоятельно.

О модулях, использованных в книге

Система Drupal находится в постоянном движении, а круг модулей, созданных ее пользователями, постоянно расширяется. В тексте книги упоминаются версии, которые имелись на момент ее написания, но время идет, и версии, доступные на сайте Drupal (*http://drupal.org*), наверняка изменятся. Одни изменения не настолько серьезны, чтобы оказать влияние на работу примеров, но другие могут оказаться более существенными.

Во многих главах практические разделы содержат примеры, которые будут сохранять свою работоспособность еще очень долгое время или изменения будут не настолько значительными, чтобы они потеряли свою актуальность. Даже если пользовательский интерфейс модуля изменится, после работы с этой книгой и знакомства с различными параметрами настройки вы сможете самостоятельно заняться исследованием модулей.

В дополнение к практическим рецептам вы познакомитесь с советами и приемами, рассказывающими о самой «кухне», то есть о том, как самостоятельно изучать модули.

Кроме того, разделы под заголовком «В центре внимания», где обсуждаются особенности модулей и проводятся сравнения с другими модулями, а также приложение B, где рассказывается, как оценить тот или иной модуль, призваны обеспечить прочный фундамент, на основе которого вы сможете самостоятельно оценивать модули. Вы сможете сравнивать новые модули по мере их появления и принимать наиболее оптимальные для пользы дела решения.

Эта книга послужит не только руководством, но и трамплином к обретению мастерства для вхождения в сообщество разработчиков проекта Drupal.

Содержимое книги

За исключением начальных глав, подготавливающих почву для последующего обсуждения, эта книга организована как последовательность рецептов, каждый из которых имеет следующую структуру:

Введение

Краткий обзор используемых модулей и общая цель главы.

Задача

В этом разделе описываются потребности вымышленного клиента, заказывающего разработку веб-сайта, который может быть сайтом wiki, фотогалереей или каталогом товаров. Здесь приводятся некоторые вводные сведения о таком клиенте и подробно рассматриваются его возможные конкретные требования.

Примечания к реализации

Здесь обсуждаются различные решения на основе Drupal, удовлетворяющие требованиям клиентов, а также детально рассказывается, какие модули были выбраны и почему. В этом разделе мы будем сравнивать и противопоставлять модули и выяснять, в каком случае лучше использовать модуль A или почему применение модуля B непродуктивно.

В центре внимания

В каждой главе представлены один-два основных модуля или концепции Drupal, а в разделах «В центре внимания» в общих чертах рассказывается о назначении каждого из них и о том, как они функционируют. Эти разделы являются своего рода «техническим паспортом», подчеркивающим особенности и возможности указанного модуля.

Практика

После описания возможностей модуля в общих чертах в практическом разделе демонстрируется, как модуль настраивается, и предоставляется пошаговый «рецепт» точного воспроизведения функциональности, необходимой клиенту.

Дополнительно

Существует масса полезных дополнительных модулей, которые могут применяться в каждом конкретном случае для достижения более широких возможностей. В этих разделах будут представлены ссылки на дополнительные модули, расширяющие функциональные возможности модулей, использованных в практических разделах.

В заключение

В этих разделах подводится итог тому, что было изучено в главе, а также даются ссылки на использовавшиеся модули и на другие ресурсы, где можно найти дополнительную информацию.

Ниже перечислены главы этой книги. Первые три главы содержат необходимые сведения для тех, кому не приходилось раньше использовать систему Drupal. Остальные главы предполагают знание основ Drupal, а также модулей Views и CCK. Если вам приходилось использовать Drupal 5, но нет опыта работы с Drupal 6, вам также может потребоваться прочитать эти главы (в частности, главу 3, так как в версии Drupal 6 модуль Views претерпел существенные изменения).

Глава 1. *Обзор системы Drupal*

Эта глава отвечает на главные вопросы о системе Drupal, ответы на которые необходимо знать: что такое система Drupal, кто и почему ею пользуется и как она действует? Здесь также приводится краткое описание истории развития Drupal, вводится основная терминология и предоставляются остальные сведения, необходимые для быстрого освоения системы.

Глава 2. *Запускаем Drupal*

Первая практическая глава, в которой описываются и демонстрируются основы использования базовой функциональности системы Drupal, а также некоторых сторонних модулей на примере создания простого веб-сайта. К концу этой главы вы должны достаточно комфортно чувствовать себя в административном разделе Drupal и знать, как добавлять простое информационное наполнение с помощью визуального интерфейса модулей FCKeditor и IMCE. Здесь также обсуждаются модули, которые помогут решить проблемы в случае неправильного обращения с сайтом, включая модуль Mollom.

Глава 3. *Каталог вакансий*

В этой главе в процессе создания веб-сайта каталога вакансий будут представлены модули Content Construction Kit (CCK) и Views. Вы узнаете, как создаются нестандартные типы информационного наполнения и добавляются поля форм и как обеспечить совместное использование списков, содержащих информационное наполнение любого типа, которые составляют основу всех последующих глав книги.

Глава 4. *Обзоры продуктов*

В этой главе мы построим веб-сайт каталога продуктов с использованием модуля Amazon для представления данных о продуктах, а также модулей Voting API и Fivestar, реализующих графические элементы отображения рейтинга.

Глава 5. *Wiki*

В этой главе рассматриваются несколько инструментов, которые могут использоваться для создания wiki в системе Drupal и для других целей. Подробно будут рассмотрены: система ревизий узлов (в паре с интересным модулем Diff), фильтр Markdown, упрощающий ввод разметки HTML, модуль Freelinking, позволяющий автоматически создавать и связывать ссылками страницы wiki, и модуль Pathauto, автоматически создающий адреса URL, удобные для поисковых систем.

Глава 6. *Управление процессом публикации*

Обсуждается организация собственного процесса публикации с помощью системы Actions, объединенной с модулем Workflow, а также рассматриваются модули Views Bulk Operations и Workspace, используемые для создания собственных административных страниц.

Глава 7. *Фотогалерея*

Эта глава поможет создать семейный фотоальбом с помощью модуля ImageField, а также расскажет о модуле ImageCache, используемом для автоматического создания миниатюр изображений заданного размера.

Глава 8. *Многоязычные сайты*

В этой главе рассказывается, как создавать многоязычные сайты, используя комплект модулей Locale, Content Translation и Internationalization.

Глава 9. *Управление событиями*

Эта глава посвящена вопросам управления событиями в системе Drupal. Демонстрируется использование модулей Date и Calendar для хранения и отображения информации о событиях и модуля Flag для сохранения информации о посетителях.

Глава 10. *Интернет-магазин*

Демонстрируется применение мощного комплекта модулей Ubercart на примере создания интернет-магазина по продаже футболок с поддержкой каталога товаров, корзины с покупками и обработки платежей.

Глава 11. *Оформление сайта*

В этой главе представлена обзорная информация о системе визуального оформления и некоторые основные приемы, которые можно использовать для изменения внешнего вида сайта Drupal. Прочитав эту главу, вы сможете приступить к изменению файлов шаблонов и придать своему сайту, созданному на основе Drupal, неповторимый внешний вид!

Приложение A. *Установка и обновление Drupal*

При отсутствии опыта работы с системой Drupal это приложение поможет установить систему и расскажет, как выполнять ее обновление в будущем.

Приложение B. *Выбор модулей*

Оценка модулей часто становится камнем преткновения при создании сайтов на базе Drupal. В этом приложении разбираются стратегии и даются советы о том, как выбрать модули для создания сайта.

Приложение C. *Модули и темы оформления, использованные в этой книге*

Перечисляются все модули и темы оформления, использовавшиеся в главах, которые позволяют самостоятельно создать код из практических разделов.

Типографские соглашения, используемые в книге

В этой книге приняты следующие типографские соглашения:

Курсив

Используется для выделения имен файлов, каталогов, новых терминов, адресов URL и важных фрагментов текста.

`Моноширинный шрифт`

Применяется для фрагментов программного кода, содержимого файлов, команд и результатов работы команд.

Таким способом выделяются советы, предложения и примечания общего характера.

Таким способом выделяются предупреждения и предостережения.

Любые перемещения по страницам Drupal оформляются, как показано ниже:

Administer→Site building→Modules (*admin/build/modules*).

В этой строке сообщается, что следует щелкнуть на ссылке Administer в блоке навигации, затем на ссылке Site building и затем на ссылке Modules. Помимо этого можно ввести путь, указанный в круглых скоб-

ках, в строку ввода URL броузера: *http://www.example.com/admin/build/modules*.

Использование программного кода примеров

Данная книга призвана помочь в решении ваших задач. Вы можете свободно использовать примеры программного кода из этой книги в своих приложениях и в документации.

Весь программный код, включая код системы Drupal 6, доступ к которому можно получить на сайте издательства O'Reilly (как, будет описано чуть ниже), распространяется на условиях GNU General Public License версии 2. Вы можете использовать программный код Drupal, включая копирование, изменение и распространение, в соответствии с условиями этой лицензии. Кроме того, название «Drupal» является торговой маркой, зарегистрированной Дризом Байтаертом (Dries Buytaert), основателем проекта Drupal. Информацию о правах на использование программного кода и торговой марки можно найти на веб-сайте Drupal (*http://drupal.org*), где также можно узнать, какие ограничения накладывает лицензия GNU General Public License на использование программного кода. Дополнительную информацию о лицензии можно найти по адресу *http://www.gnu.org/licenses/old-licenses/gpl-2.0.html#SEC3*.

В отношении программного кода примеров из этой книги вам не нужно обращаться в издательство за разрешением, если вы не собираетесь воспроизводить существенные части программного кода, не являющегося составной частью Drupal. Например, если вы разрабатываете программу и используете в ней несколько отрывков программного кода из книги, вам не нужно обращаться за разрешением. Однако в случае продажи или распространения компакт-дисков с примерами из книг издательства O'Reilly вам необходимо получить разрешение от издательства. Для цитирования данной книги или примеров из нее при ответе на вопросы не требуется получение разрешения. При включении существенных объемов программного кода примеров из этой книги в вашу документацию вам необходимо будет получить разрешение издательства.

Мы приветствуем, но не требуем добавлять ссылку на первоисточник при цитировании. Под ссылкой на первоисточник мы подразумеваем указание авторов, издательства и код ISBN книги. Например: «Using Drupal by Angela Byron, Heather Berry, Nathan Haug, Jeff Eaton, James Walker, and Jeff Robbins. Copyright 2009 Angela Byron, Heather Berry, Nathan Haug, Jeff Eaton, James Walker, and Jeff Robbins, 978-0-596-51580-5».

За получением разрешения на использование значительных объемов программного кода примеров из этой книги обращайтесь по адресу *permissions@oreilly.com*.

Загрузка Drupal 6

На веб-сайте этой книги по адресу *http://usingdrupal.com/source_code* приводятся ссылки для загрузки копии Drupal 6, а также всех модулей, рассматриваемых в этой книге, и тем оформления, задействованных в примерах веб-сайтов в практических разделах глав. В каждом практическом разделе главы также имеется «профиль установки» (набор сценариев, выполняющих начальное определение параметров настройки по умолчанию), который воспроизводит начальный сайт. Эти профили установки могут задействоваться в начале процесса установки Drupal, например «Глава 4: каталог вакансий».

Для перехода от примеров из практического раздела одной главы к примерам из другой необходимо создать новый сайт, используя тот же исходный программный код. Сделать это можно с минимальными усилиями, выполнив следующие шаги:

1. Создать новую базу данных для установки Drupal для нужной главы или удалить и вновь создать существующую базу данных.
2. Скопировать файл *sites/default/default.settings.php* в файл *sites/default/settings.php*, перезаписав существующий файл *settings.php*.
3. Изменить права доступа к файлу *sites/default/settings.php*, чтобы он стал доступен для записи.
4. Запустить сценарий установки *http://www.example.com/install.php*.

Дополнительная информация об установке Drupal приводится в приложении А.

Помимо настройки некоторых базовых параметров, таких как имя сайта, выбор темы оформления и других, профили установки в каждой главе (кроме главы 2) также настраивают следующие учетные записи:

имя пользователя: *admin*, пароль: *oreilly*

Первый пользователь, которому дается роль «администратора сайта». Этот пользователь может выполнять на сайте любые действия.

имя пользователя: *editor*, пароль: *oreilly*

Пользователь, которому дается роль «редактора»; используется в главах, где требуется наличие пользователей с повышенными правами доступа.

имя пользователя: *user*, пароль: *oreilly*

Обычный пользователь, которому дается роль «зарегистрировавшего пользователя».

Это и есть те типы пользователей, на которые инструкции ссылаются как на пользователя, зарегистрированного с правами «editor» или с иными. Если не указано иное, предполагается, что действия должны выполняться с правами пользователя «admin».

Safari® Books Online

Если на обложке книги есть пиктограмма «Safari® Books Online», это означает, что книга доступна в Сети через O'Reilly Network Safari Bookshelf.

Safari предлагает намного лучшее решение, чем электронные книги. Это виртуальная библиотека, позволяющая без труда находить тысячи лучших технических книг, вырезать и вставлять примеры кода, загружать главы и находить быстрые ответы, когда требуется наиболее верная и свежая информация. Она свободно доступна по адресу *http://safari.oreilly.com*.

Отзывы и предложения

С вопросами и предложениями, касающимися этой книги, обращайтесь в издательство:

O'Reilly Media
1005 Gravenstein Highway North
Sebastopol, CA 95472
800-998-9938 (в Соединенных Штатах Америки или в Канаде)
707-829-0515 (международный)
707-829-0104 (факс)

Список опечаток, файлы с примерами и другую дополнительную информацию вы найдете на сайте книги:

http://www.oreilly.com/catalog/9780596515805

Свои пожелания и вопросы технического характера отправляйте по адресу:

bookquestions@oreilly.com

Дополнительную информацию о книгах, обсуждения, Центр ресурсов издательства O'Reilly вы найдете на сайте:

http://www.oreilly.com

Благодарности

Команда Lullabot выражает свою благодарность техническим редакторам книги, включая Роберта Дугласа (Robert Douglass), Аджая Галлвейла (Ajay Gallewale), Джеффри Макгуайра (Jeffrey MacGuire), Дэвида Мура (David Moore) и Мэтта Уэстгейта (Matt Westgate). Благодарим Татьяну Апанди (Tatiana Apandi) и Джулию Стил (Julie Steele) – сотрудников издательства O'Reilly, руководивших процессом создания нашей первой коллективной книги. Мы также благодарим наших коллег Лизу Киндред (Liza Kindred), Хейли Скарпино (Haley Scarpino) и Тима Мак Дормана (Tim McDorman), которые помогли нам скорректировать рабо-

чий график так, чтобы мы могли закончить эту книгу. Спасибо Джеффу Итону (Jeff Eaton), предоставившему фотографии для главы, описывающей создание сайта фотогалереи. Также спасибо Кенту Байю (Kent Bye) из Lullabot за разработку визуализации в модуле Views и Джону Вандику (John VanDyk) за его бесценные советы по некоторым главам. И особое спасибо Ивану Зугеку (Ivan Zugec), любезно предоставившему нам домен *usingdrupal.com*. И, конечно же, спасибо Дризу Байтаерту (Dries Buytaert) за изобретение и открытость Drupal – без него все это было бы невозможно.

Анжела Байрон (Angela Byron) в первую очередь выражает горячую благодарность своей спутнице Марси Маккей (Marci McKay), проявлявшей необычайное терпение и понимание по отношению к работе за полночь и чрезвычайно терпимо и даже благосклонно относящейся к навязчивой идее Анжелы, каковой является для нее система Drupal. Также огромное спасибо она говорит своей семье – матери Дженни и отцу Майку, сестрам Кейт и Саре – за их поддержку в процессе работы над книгой. Джон Уайт (John Wait) и Дебра Уильямс-Каули (Debra Williams-Cauley) также заслуживают благодарности за их помощь воплотить мечту Анжелы написать книгу. Майкл Кокс (Michelle Cox) и Мэтью Харрисон (Matthew Harrison) помогли выполнить первую «чистку» рукописи книги. Моше Вейтцман (Moshe Weitzman), Брендон Бергрен (Brandon Bergren) и Дриз Байтаерт (Dries Buytaert) выполнили техническое редактирование некоторых ранних версий глав книги, при этом Дриз особенно вдумчиво отнесся к этому и во всем поддерживал Анджелу.

Эддисон Берри (Addison Berry) хотела бы выразить благодарность своему супругу Коллину Макграву (Colleen McGraw), который стоически относился к пропадающим выходным и отсутствию помощи по дому и вдохновлял ее во всех жизненных перипетиях. Ричард Барфорд (Richard Burford), Алекс Дергачев (Alex Dergachev), Джоэл Фаррис (Joel Farris), Джей Макдональд (Jay McDonald), Дон Палмер (Don Palmer), Хосе Рейеро (Jose Reyero) и Брайан Вуик (Brian Vuyk) любезно дали свои отзывы к ее главам, а Уим Лирс (Wim Leers) оказал существенную помощь в переводе на голландский язык. Все сообщество Drupal всемерно поддерживало ее в постижении Drupal и в создании этой книги. И то, и другое было бы невозможно без их помощи. Наконец, она говорит спасибо своим родителям Джоан и Мерлин Берри за поддержку всех ее замыслов и веру в нее.

Натан Хог (Nathan Haug) хотел бы поблагодарить своих замечательных родителей Джеймса и Аледу Хог, а также воодушевлявшего его деда Тома Анберга. Он благодарен техническим редакторам Дэвиду Муру (David Moore) и Брайану Вуику (Brian Vuyk), а также выражает особую признательность всем создателям платформы Drupal и дополнительных модулей. Нат благодарит Эрла Милза (Earl Miles) за помощь с модулем Views, Джонатана Чаффера (Jonathan Chaffer), Карен Стивенсон (Karen Stevenson) и Ив Чедмуа (Yves Chedemois) за помощь

с модулем CCK. Платформа Drupal никогда бы не стала тем, что она представляет собой сейчас, без эффективного сотрудничества многих личностей.

Джефф Итон (Jeff Eaton) хотел бы поблагодарить свою супругу Кэтрин за величайшее долготерпение и поддержку. Романтический ужин не включает в себя тестирование модулей и вылавливание неполадок, и следует признать, что то, с каким юмором и пониманием отнеслась Кэтрин к вторжению высоких технологий в семейную жизнь, сыграло важную роль в доведении этого проекта до конца. Дуг Грин (Doug Green), Эрл Майлз (Earl Miles) и многие члены сообщества Drupal давали превосходные советы и указывали на сложности, которые легко могли бы сбить с толку новых пользователей. Джейсон Скотт (Jason Scott) и Джефф Бенсон (Jeff Benson) проявляли бесконечное сочувствие и поздней ночью, и ранним утром, а родители Джеффа, Дуг и Синди, не пожалели лучшей части двух десятков лет, чтобы оказать поддержку в его увлечениях.

Джеймс Уокер (James Walker) очень признателен своим детям Эндрю и Камрин за их любовь, терпение и прогулки в парке. Карен Стивенсон (Karen Stevenson), Райан Шрама (Ryan Szrama), Эрл Майлз (Earl Miles) и Нат Хог (Nate Haug) помогали ему с программным кодом, который оказался чрезвычайно полезным для раскрытия сути вопросов, поднимаемых в его главах. Он благодарен своей маме Линде, которая всегда верила в него и давала пристанище для Lullabot.

Джефф Роббинс (Jeff Robbins) благодарен своей супруге Дженнифер за ее любовь, поддержку и редакторское мастерство и сыну Арло за его умение все превращать в увлечение. Спасибо издательству O'Reilly Media, которое создавало ощущение большой и дружной семьи в 90-х годах, когда Джефф и Дженнифер там работали и где они познакомились. Джефф хотел бы выразить свою благодарность Мэтту Уэстгейту (Matt Westgate), который стал его лучшим, насколько это только можно вообразить, деловым партнером и, кроме того, обеспечил атмосферу надежности, безмятежности и благополучия, в которой процветала компания Lullabot и ее сотрудники. Джефф также выражает свою благодарность и признательность Лайзе Киндред (Liza Kindred), которая обеспечивала работу компании, и остальным членам команды, у которых всегда находилось, чем поразить его, а также все сообщество пользователей Drupal – за щедрость и изумительный программный код, который мы и называем Drupal.

1

Обзор системы Drupal

Эта книга покажет вам, как создаются самые разнообразные веб-сайты с использованием веб-платформы Drupal. Независимо от того, занимаетесь вы продвижением своей рок-группы или создаете сайт в локальной сети компании, некоторые решаемые задачи окажутся общими для всех случаев. Одно из базовых понятий – *информационное наполнение*, или *содержимое*, которым обладает любой сайт; это могут быть аудиоматериалы, текст, анимированные изображения в формате GIF, и веб-сайт представляет это содержимое окружающему миру. Кроме того, необходимо реализовать *управление* этим информационным наполнением. Обладая достаточными знаниями основных веб-технологий, вы можете создать такую систему самостоятельно, однако система Drupal существенно облегчит создание веб-сайта, а также расширение его возможностей и повседневное добавление, редактирование и удаление содержимого. Наконец, у вашего сайта будут *посетители*, а в этой книге представлены разнообразные способы, основанные на использовании Drupal, применяя которые вы сможете привлекать пользователей и взаимодействовать с ними.

Эта глава начинается с основных сведений о системе Drupal: что это такое, кто ею пользуется, почему разработчики выбирают именно эту систему. Затем приводится обзор, который начинается с объяснения того, что понимается под неоднозначным термином «управление содержимым» и как с применением этой концепции создаются веб-сайты. А в заключение будут даны определения и описания основных концепций Drupal, знание которых необходимо для понимания принципов обработки содержимого в системе Drupal.

Что такое Drupal?

Drupal – это открытая *система управления содержимым* (Content Management System, CMS), которая используется сотнями тысяч органи-

заций и отдельных людей для создания привлекательных веб-сайтов с богатым информационным наполнением.[1] Создание веб-сайта на основе Drupal заключается в комбинировании различных «строительных блоков», о которых будет говориться ниже в этой главе и которые позволяют точно настроить функциональность конкретного веб-сайта. Будучи один раз создан, веб-сайт на платформе Drupal позволяет обеспечить управление посредством форм без необходимости что-то перепрограммировать вручную. Система Drupal распространяется бесплатно; она обладает огромной библиотекой постоянно развивающихся инструментов, которые могут использоваться для улучшения веб-сайта.

Drupal также является *платформой управления содержимым* (Content Management Framework, CMF). В дополнение к инструментам построения веб-сайтов для веб-мастеров она предоставляет программистам и разработчикам возможность настраивать сайты с помощью дополнительных модулей расширения, которые позволяют регулировать поведение практически всех элементов Drupal. В мире существуют тысячи модулей, реализующих функциональные возможности, от фотогалерей до корзин с покупками и переводчиков на «пиратский язык». Большинство модулей было передано сообществу Drupal, они доступны для загрузки и могут использоваться в вашем сайте, построенном на базе Drupal. Все функциональные возможности, рассматриваемые в этой книге, реализованы за счет объединения «ядра» Drupal и модулей, созданных сообществом.

Особого упоминания заслуживает *сообщество* Drupal; человеческий фактор в системе Drupal обоснованно считается одним из самых важных и ценных ее активов. На момент выхода версии Drupal 6 в феврале 2008 года в разработке основных программных компонентов принимали участие более 700 членов сообщества. В сопровождении модулей было задействовано более 2000 разработчиков, еще больше было занято в тестировании, документировании, поддержке пользователей, переводах и других важных областях проекта. Те, кто понимает значимость открытых платформ, подтвердят важность наличия активно живущего сообщества.

Кто пользуется Drupal?

На протяжении последних лет популярность Drupal росла взрывообразно, что не могли не заметить некоторые крупные компании. Многие медиакомпании, такие как MTV UK, Lifetime и Sony BMG Records, используют Drupal как основу для создания сообществ вокруг своих продуктов. Такие издательства, как New York Observer и The Onion,

[1] Дополнительную информацию о движении за распространение программного обеспечения с открытыми исходными текстами можно найти на сайте *http://opensource.org*, который по случайности также построен на основе системы Drupal.

журналы *Popular Science* и *Fast Company* используют Drupal для предоставления своим читателям интерактивного информационного наполнения. Организации Amnesty International, Организация Объединенных Наций и Electronic Frontier Foundation используют Drupal для координации своих действий, направленных на решение важных проблем. Открытые проекты Ubuntu Linux, Eclipse, Firefox и jQuery используют Drupal для организации своих сообществ разработчиков. Такие блогеры, как Тим Бернерс-Ли (Tim Berners-Lee), Хизер Б. Армстронг (Heather B. Armstrong) (известный также как Dooce), сообщество BlogHer и Мерлин Манн (Merlin Mann) используют Drupal как платформу для своих публикаций. На рис. 1.1 показаны некоторые из широко известных веб-сайтов, построенных на базе Drupal.

Все эти сайты объединяет потребность в мощных средствах публикации и высокая коммуникативность.

Рис. 1.1. *Главные страницы наиболее известных веб-сайтов, построенных на базе Drupal*

В Интернете существует несколько мест, где можно получить дополнительную информацию о том, кто на сегодняшний день использует Drupal. Дриз Байтаерт (Dries Buytaert), основатель проекта Drupal, ве-

дет список наиболее известных веб-сайтов, построенных на базе Drupal, в своем блоге *http://buytaert.net/tag/drupal-sites*. На веб-сайте проекта Drupal имеется раздел, содержащий подробные исследования различных ситуаций и истории успеха (*http://drupal.org/cases*). Кроме того, по адресу *http://www.drupalsites.net* находится каталог, содержащий тысячи веб-сайтов на платформе Drupal, найденных в Интернете, – от маленьких персональных веб-сайтов до крупных сайтов социальных сетей с миллионами активных пользователей.

Возможности, которые поддерживает Drupal

Система Drupal предоставляет весьма широкий круг возможностей, которые более подробно обсуждаются в главе 2. В их число входят:

Гибкая модульная система

Модули и расширения, способные изменять существующие и добавлять новые возможности в сайты на базе Drupal. Практически для любой функциональности имеется либо точно реализующий ее модуль, либо она может быть реализована путем объединения нескольких модулей, либо существует некоторый программный код, который способен послужить основой реализации.

Настраиваемая система визуального оформления

Вывод любой информации в Drupal может настраиваться в широких пределах, благодаря чему вы можете привести внешний вид своего сайта в соответствие со своими желаниями (или, точнее, в соответствие с желаниями дизайнера).

Расширенные возможности создания информационного наполнения

Вы можете определять новые типы информационного наполнения (блог, событие, фраза дня) на лету. Дополнительные модули дают возможность шагнуть в этом направлении еще дальше и позволяют администраторам создавать собственные поля внутри вновь созданных типов содержимого.

Встроенная оптимизация под поисковые системы

Система Drupal предлагает «коробочную» поддержку системы удобочитаемых адресов URL, а вся выходная информация Drupal соответствует стандартам. Обе эти особенности обеспечивают дружественность веб-сайтов по отношению к поисковым системам.

Система прав доступа на основе ролей

Собственные роли и изобилие различных разрешений дают возможность организовать высокую избирательность прав доступа к компонентам системы. А существующие модули способны повысить эту избирательность еще больше, вплоть до уровня каждого отдельного пользователя.

Средства общения и совместной работы

Система Drupal имеет такие встроенные инструменты, как блоги, поддержка возможности оставлять комментарии, форумы и настраиваемые профили пользователей. Чтобы добавить практически любую возможность, какую только можно себе представить, например систему рейтингов, группировку пользователей или инструменты модерирования, достаточно лишь загрузить соответствующий модуль.

Краткая история систем управления содержимым

Прежде чем перейти к более близкому знакомству с системой Drupal, совершим небольшой экскурс в прошлое, в дни, предшествовавшие появлению систем управления содержимым. Чтобы понять, как действуют Drupal и другие CMS, упрощающие нашу работу, мы посмотрим, как была организована работа сайтов во времена молодости Интернета.

Исторический экскурс в процесс создания веб-сайта

В туманном прошлом (в 1990-х годах, для тех, кто помнит цеппелины и автомобили Форд-Т) веб-страницы были ни чем иным, как простыми текстовыми файлами, сложенными в папки на сервере где-то в недрах Интернета. Эти файлы, с такими именами, как *index.html*, *news.html*, *about_us.html* и так далее, были доступны любому для просмотра с помощью веб-броузера. Используя язык разметки HTML, эти файлы могли ссылаться друг на друга, включать информацию в виде изображений и в других форматах и вообще представлять себя. *Веб-сайты* тех дней, как джинсы в обтяжку, представляли собой всего лишь набор таких файлов в определенном каталоге, как показано на рис. 1.2.

Такая система прекрасно зарекомендовала себя, и она не была лишена смысла. Каждому адресу URL в Интернете, который мог посетить пользователь, соответствовал уникальный файл *.html*, находящийся на веб-сервере. Если возникала необходимость организовать их в виде разделов, просто создавался новый каталог и соответствующие файлы перемещались в этот каталог, например адрес *http://www.example.com/news/* мог быть адресом раздела «Новости» на сайте, а информационный бюллетень за 1997 год мог находиться по адресу *http://www.example.com/news/fall_1997_products.html*. Когда веб-мастеру (или другому специалисту) требовалось исправить какую-либо проблему, он мог просто просмотреть страницу с помощью своего веб-броузера, а затем открыть соответствующий файл на веб-сервере и исправить его.

К сожалению, по мере увеличения веб-сайтов в размерах стало очевидно, что такой подход плохо поддается масштабированию. Примерно че-

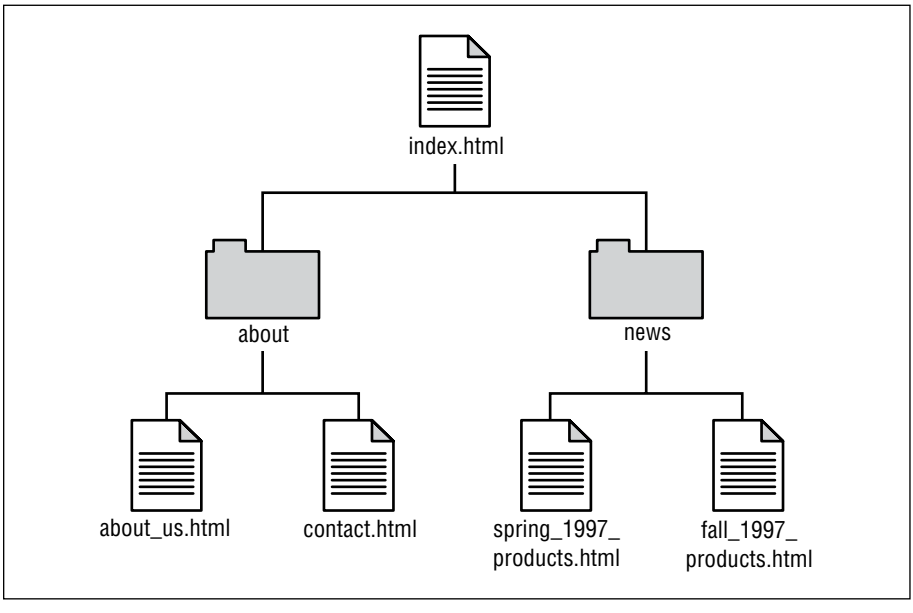


Рис. 1.2. *Традиционная структура веб-сайта 90-х годов*

рез год добавления новых страниц и перетасовки их по каталогам многие сайты могли насчитывать десятки, сотни, а иногда даже тысячи страниц. И это, друзья мои, приводило к появлению серьезных проблем:

Любое изменение дизайна сайта требовало выполнения огромного объема работ.

Форматирование информации, ее размещение и другие дизайнерские творческие решения воплощались в каждой отдельной странице. Каскадные таблицы стилей (Cascading Style Sheets, CSS) еще не взяли Интернет штурмом, поэтому такие простые задачи, как изменение шрифта, используемого сайтом по умолчанию, требовали (что совершенно очевидно) внести изменения в *каждый отдельно взятый файл*.

Структура сайта приводила к множественному дублированию содержимого.

Большинство проектов веб-сайтов предусматривало наличие на страницах стандартного нижнего колонтитула с упоминанием авторских прав и контактной информации, изображения в заголовке или навигационного меню некоторого вида и так далее. Если что-то изменялось, приходилось вносить изменения во все файлы. Если какому-то веб-мастеру очень повезло и все, кто работал до него, очень добросовестно относились к своему делу, то изменения можно было реализовать с помощью сценария. Но большинству не повезло, и они по сей

день с ужасом вспоминают сайты, построенные с помощью FrontPage, PageMill, Dreamweaver или Notepad.

Было невозможно обеспечить единообразие и своевременное обновление сайтов.

Большинство сложных веб-сайтов было организовано в виде каталогов и подкаталогов, что обеспечивало разумную структуру. Добавление новой статьи в каталог *news* подразумевало необходимость изменить страницу «краткий обзор новостей», где перечислены все статьи, и, возможно, дополнить краткой ссылкой на главной странице веб-сайта, и (ужас!) не забыть переместить ссылку вниз, когда новость перестанет быть «свежей». Крупный сайт с множеством разделов и с большим объемом содержимого одними такими обновлениями мог обеспечить полную занятость веб-мастера.

Эпоха сценариев и баз данных

Поиск решения этих проблем привел к первой настоящей революции в веб-дизайне: использованию сценариев и программ *CGI* (*Common Gateway Interface – общий интерфейс шлюза*). Первый шаг заключался в использовании специальных тегов с именем «*вставка на стороне сервера*» (*Server-Side Includes, SSI*) в каждом файле HTML. Эти теги позволили веб-дизайнерам вставлять содержимое одного файла (например, стандартную информацию об авторских правах или список последних новостей) в текущую страницу на стороне сервера, как если бы оно было частью самого файла HTML. Это существенно упростило обновление фрагментов информации, так как теперь они хранились в одном месте.

Второй шаг заключался в использовании простых баз данных для хранения однотипных фрагментов содержимого. Все новости на сайте CNN.com (*http://www.cnn.com*) имеют похожую структуру, даже если их содержимое отличается. То же справедливо для всех страниц с описаниями продуктов на сайте Apple.com (*http://www.apple.com*), для всех записей на сайте Blogger.com (*http://www.blogger.com*) и так далее. Вместо того чтобы хранить каждый фрагмент в отдельном файле HTML, веб-мастера стали использовать программы, работающие на стороне веб-сервера, которые отыскивали содержимое каждой статьи в базе данных и отображали его, обертывая разметкой HTML, соответствующей макету сайта. Такие адреса URL, как *http://www.example.com/news/1997/big_sale.html*, заменились адресами типа *http://www.example.com/news.cgi?id=10*. Теперь вместо того чтобы отыскивать в каталоге *news* и в подкаталоге *1997* и возвращать файл *big_sale.html*, веб-сервер должен был запустить программу *news.cgi*, позволить ей извлечь из базы данных статью с номером 10 и вернуть броузеру пользователя текст, который выведет запущенная программа.

Все эти отличия требовали изменения подходов к созданию веб-сайтов. Но преимущества были более чем очевидны: десятки и даже сотни фай-

лов можно было заменить одним или более сценариями, управляемыми базой данных, как показано на рис. 1.3.

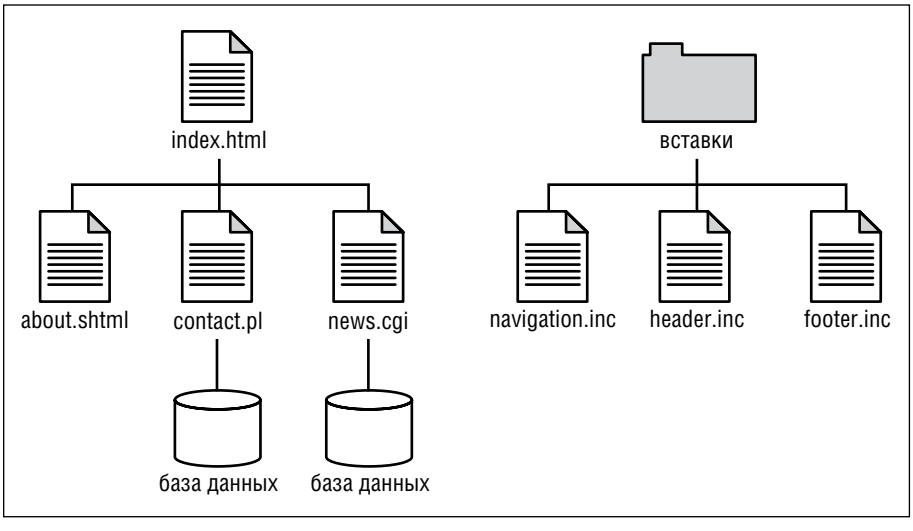


Рис. 1.3. *Переход от отдельных файлов к использованию сценариев, управляемых базами данных*

Но даже с этими улучшениями оставались нерешенными серьезные проблемы:

Где следует вносить изменения?

Крупные сайты с множеством разнообразных типов информационного наполнения (информация о продуктах, биографии служащих, пресс-релизы, файлы для загрузки и так далее) все еще состояли из разнообразных сценариев, отдельных баз данных и других элементов, которыми требовалось управлять. Веб-мастеру, выполняющему обновление, необходимо было определить, следует ли изменять файл HTML, запись в базе данных или программный код сценария.

Слишком большое число маленьких фрагментов, собранных вместе

Динамическое содержимое, такое как форумы или гостевые книги, с которым посетители могли взаимодействовать, потребовало организации собственной инфраструктуры, причем нередко каждая из этих систем проектировалась отдельно. Объединение их в общий веб-сайт было далеко не самой простой задачей.

Революция в управлении содержимым

Постепенно появились программы, предназначенные для управления этими различными видами содержимого, и возможность использовать единообразный пользовательский интерфейс. Прежние поколения про-

граммного обеспечения концентрировались на выполнении определенной задачи, тогда как новейшие реализации CMS представляют собой унифицированные инструменты для создания, редактирования и организации информационного наполнения веб-сайта. Кроме того, большинство систем предоставляют разработчикам механизмы создания расширений и реализации новых функциональных возможностей, не заставляя их при этом заново изобретать колесо. На рис. 1.4 показано, как система управления содержимым, использующая единственную базу данных и сценарий, совмещает все эти возможности.

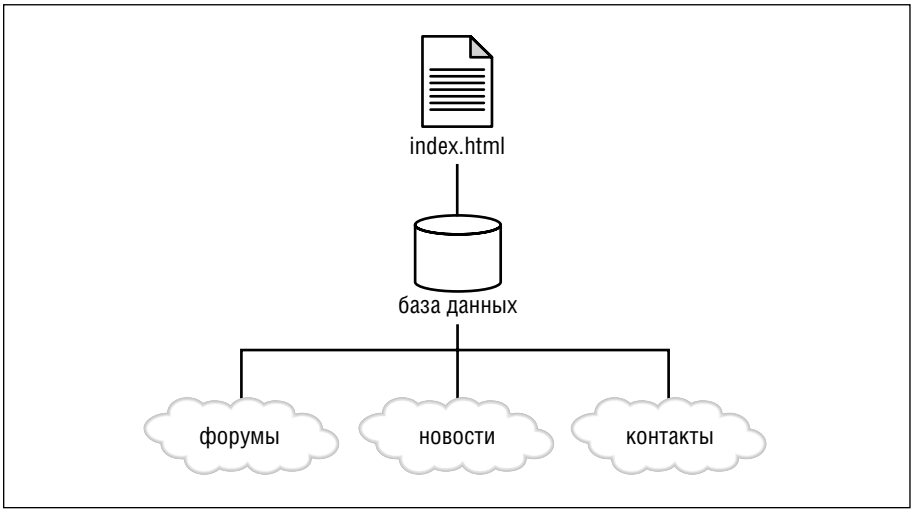


Рис. 1.4. *Структура интегрированного веб-сайта, управляемого базой данных*

Drupal – это одна из таких систем управления содержимым следующего поколения. Она позволяет создавать и организовывать содержимое самых разных видов, предоставляя инструменты управления содержимым как для эксплуатационного персонала, так и для посетителей сайта, и открывает доступ к тысячам сторонних расширений, добавляющих новые возможности. Дриз Байтаерт (Dries Buytaert), основатель проекта Drupal, сказал в своей речи на саммите Open Source CMS Summit в 2007 году, что главная цель системы Drupal в том, чтобы «убрать веб-мастера». Если вы *являетесь* веб-мастером, эти слова могут показаться угрожающими, но стоит немного поразмыслить, и они приобретут совершенно иной, захватывающий дух смысл. При использовании системы Drupal исчезает трудоемкая необходимость в организации и обновлении тысяч страниц: вы можете сосредоточиться на реализации особенностей, необходимых вашему сайту и учитывающих желания ваших пользователей.

Как работает Drupal

С концептуальной точки зрения *стек Drupal* выглядит, как показано на рис. 1.5. Drupal – это своего рода промежуточный уровень между внутренней реализацией (обеспечивающей работу Интернета) и интерфейсом (который видят посетители в окне веб-броузера).

На нижних уровнях находятся операционная система, веб-сервер, база данных и PHP. *Операционная система* обеспечивает работоспособность веб-сайта: здесь решаются такие низкоуровневые задачи, как обработка сетевых подключений, файлов и разграничение прав доступа к файлам. *Веб-сервер* обеспечивает доступность вашего компьютера в Интернете и служит для предоставления требуемой информации, когда посетитель выполняет переход по адресу *http://www.example.com*. *База данных* хранит *данные*: все содержимое веб-сайта, учетные записи пользователей и параметры настройки – в одном месте. А *PHP* – это язык программирования, который динамически генерирует страницы и перемещает информацию от базы данных к веб-серверу.

Сама система Drupal также состоит из множества слоев. На самом нижнем уровне она реализует дополнительные возможности поверх PHP, добавляя некоторые подсистемы, такие как обработка сеансов пользователей и механизм аутентификации, фильтр подсистемы безопасности и подсистема шаблонов отображения. Над этим слоем находится уровень дополнительных функциональных возможностей, называемых модулями, которые будут обсуждаться в следующем разделе этой главы. Модули расширяют возможности Drupal и могут воспроизводить содержимое любой страницы. Но, прежде чем страница появится перед пользователем, она пройдет через *систему тем оформления*, которая позволяет модифицировать оформление страниц и способна удовлетворить запросы самых взыскательных дизайнеров. Подробнее о системе тем оформления рассказывается в главе 11.

Система тем выводит содержимое страницы, обычно в формате XHTML, хотя поддерживаются и другие форматы. Для управления расположением элементов, цветом и шрифтами для заданной страницы используются каскадные таблицы стилей, а работоспособность динамических элементов интерфейса, таких как сворачиваемые наборы полей в формах и буксируемые строки таблиц в административном интерфейсе Drupal, обеспечивается с помощью JavaScript.

Мы рассказали вам о «старом» способе создания веб-сайтов с использованием статических файлов HTML, о переходе к использованию коллекций сценариев и о «новом» способе: создании полнофункциональных веб-приложений, которые управляют целыми веб-сайтами. Этот третий способ, реализованный системой Drupal, требует использовать новую концепцию строительных блоков. Они будут использоваться любым веб-сайтом, построенным на основе системы Drupal!

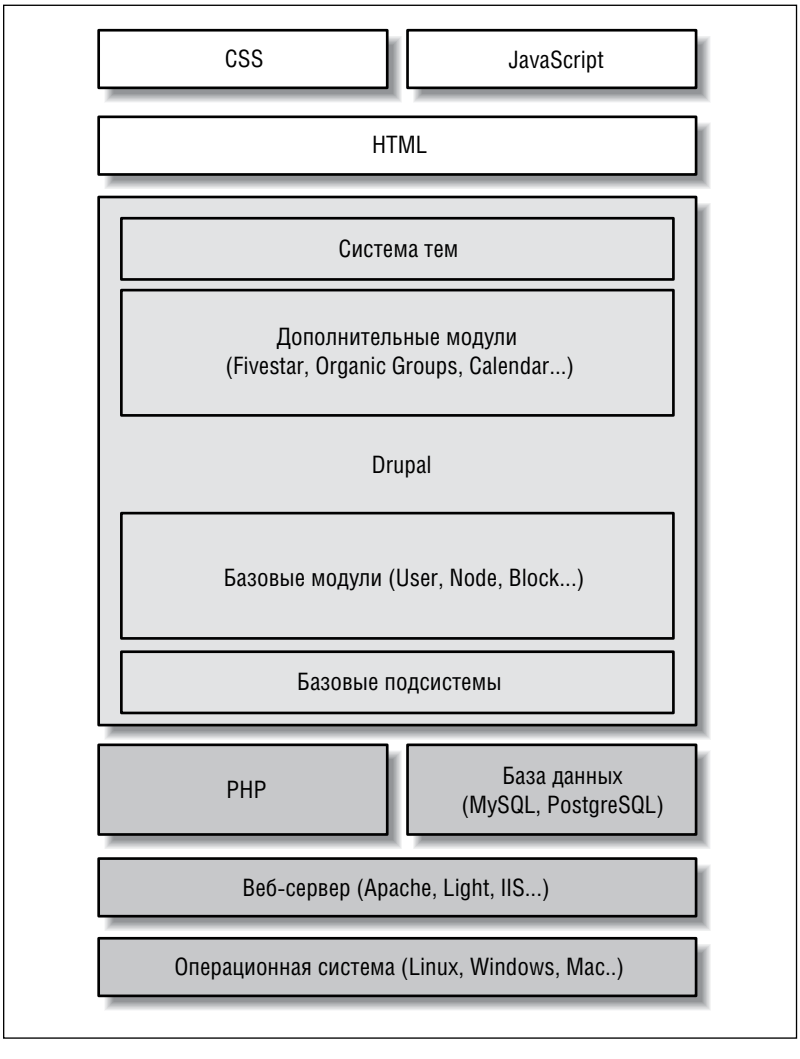


Рис. 1.5. *Расположение концептуальных уровней системы Drupal относительно других уровней веб-сайта*

Модули

Практически все в системе Drupal вращается вокруг понятия *модуля*, который является файлом с программным кодом на языке PHP, реализующим функциональные возможности, с которыми система Drupal умеет работать. Все функциональные возможности Drupal, доступные администратору или конечному пользователю, от фундаментальных особенностей, таких как возможность выполнять вход в систему, до-

бавлять новое содержимое в динамические фотогалереи и сложные системы голосования, – все они реализованы в виде модулей. Примерами модулей могут служить: модуль Contact, который реализует форму с контактной информацией, и модуль User, который реализует процедуру аутентификации пользователя и проверку прав доступа. В других системах управления содержимым модули также называются *дополнениями* или *расширениями*.

Существует два типа модулей: «базовые» модули, которые включены непосредственно в состав системы Drupal, и «сторонние» модули, которые создаются сообществом Drupal и могут загружаться и устанавливаться по отдельности. Кроме нескольких обязательных базовых модулей все остальные модули могут подключаться и отключаться, в зависимости от потребностей веб-сайта.

Несмотря на то, что существуют модули с функциональностью «под ключ», тем не менее разработчики из сообщества Drupal обычно отдают предпочтение разработке модулей, которые выполняют единственную функцию, но делают это хорошо и при этом допускают возможность комбинирования с другими модулями. Это означает, что у вас имеется практически неограниченный контроль над внешним видом и поведением вашего веб-сайта. Возможности вашей галереи изображений не ограничены представлениями разработчика о том, как она должна выглядеть и как действовать. Вы можете добавить систему рейтингов или возможность оставлять комментарии, при желании – сортировать изображения по типу фотокамеры, а не по дате. Однако для достижения этой гибкости вам придется «построить» функциональность в системе Drupal, объединив несколько различных модулей и выполнив настройку их параметров, а не просто щелкнуть на флажке «галерея изображений» и оставить его отмеченным. Мощь системы Drupal порождает сложность ее изучения, с которой вы не столкнетесь в других пакетах CMS, а изобилие доступных модулей может отпугнуть при попытке определить, чем же стоит воспользоваться в конкретном случае. В приложении B приводятся советы и рекомендации, которые помогут оценить качество модуля и его необходимость для вашего проекта.

Пользователи

Следующий строительный блок веб-сайта на базе системы Drupal – это понятие *пользователя*. В случае простого веб-сайта брошюрного типа, который будет сопровождаться единственным администратором и посещаться только потенциальными клиентами, можно создать единственную учетную запись пользователя – для самого администратора. При создании сайтов, подразумевающих общение членов сообщества, вам придется настроить систему Drupal, чтобы позволить всем желающим пользоваться сайтом регистрироваться на нем и создавать свои учетные записи.

Первый пользователь, который создается при построении нового сайта на базе Drupal, пользователь №1, – это специальный пользователь. Подобно пользователю root на сервере UNIX, пользователь №1 обладает правом выполнять любые действия на сайте Drupal. Поскольку для этого пользователя не выполняются обычные проверки безопасности, он легко может по неосторожности уничтожить информационное наполнение или как-то иначе нарушить работу сайта, если позволить использовать эту учетную запись в повседневной работе. Поэтому лучше использовать эту учетную запись для выполнения специальных административных задач и настройки, а для работы с содержимым создать дополнительную учетную запись.

Каждому добавляемому пользователю может быть присвоена настраиваемая *роль*, такая как «редактор», «платный клиент» или «важная персона». Каждой роли могут быть присвоены *права* на выполнение различных действий на веб-сайте: посещение определенных страниц, просмотр содержимого определенного вида, разрешение оставлять комментарии к существующему содержимому, заполнять профиль пользователя и даже создавать новые учетные записи и управлять их правами доступа. По умолчанию в системе Drupal имеются две предопределенные роли: *зарегистрированный пользователь* и *анонимный пользователь*. Любой создаваемой учетной записи на сайте автоматически присваивается роль «зарегистрированный пользователь», а любому посетителю, для которого еще не создана учетная запись (или который еще не прошел процедуру аутентификации со своим именем пользователя и паролем), присваивается роль «анонимный пользователь».

Содержимое (узлы)

Узлы – это следующий строительный блок в системе Drupal, причем один из самых важных. Одна из важнейших составляющих проектирования любого сайта на базе Drupal заключается в выяснении, с какими конкретными видами содержимого (которые в Drupal называются «типами содержимого») вам предстоит работать. Практически в каждом конкретном случае будут присутствовать узлы различных видов.

Все узлы независимо от типов содержимого, хранящегося в них, обладают несколькими общими базовыми свойствами:

- Автор (пользователь сайта, создавший содержимое)
- Дата создания
- Заголовок
- Тело содержимого

Вам требуется создать страницу с конфиденциальными сведениями о вашей компании? Это узел. Вам требуется дать пользователям возможность оставлять сообщения в блоге? Каждое такое сообщение – это

узел. Пользователи будут оставлять ссылки на интересные материалы, размещенные где-то в Интернете? Каждая такая ссылка, как и следовало ожидать, будет храниться как узел.

В дополнение к общим базовым свойствам все узлы могут обладать определенными особенностями, встроенными в систему Drupal, такими как флаги, указывающие, будут ли публиковаться данные узлы, и настройки, управляющие способом отображения узлов каждого типа. Права на создание и редактирование содержимого узлов каждого типа также могут быть присвоены различным пользовательским ролям, например пользователи с ролью «blogger» могли бы создавать содержимое с типом «Blog entry» (Сообщение в блоге), но только пользователи с ролями «administrator» (администратор) или «editor» (редактор) могли бы создавать узлы с типом «News» (Новости).

Узлы могут также хранить информацию о ревизиях, описывающих каждое изменение, внесенное с момента создания узла. Если вы допустили ошибку (например, удалив важный параграф на странице «About Us» (о компании)), это позволит легко восстановить предыдущую версию узла.

В состав системы Drupal входят два предопределенных типа узлов: «Page» (Страница) и «Story» (Статья). Они не играют какую-то специальную роль – они просто предлагают стандартный набор особенностей для всех узлов и ничего больше. Единственное отличие между этими двумя типами узлов заключается в параметрах настройки по умолчанию. Узлы «Page» не отображают информации об авторе или дате создания. Они отлично подходят для размещения такого содержимого, как «About Us» (О компании) и «Terms of Service» (Условия обслуживания), у которого не может быть определенного автора. Узлы типа Story отображают эту информацию, а кроме того, настроены так, чтобы появляться на главной странице сайта в случае их публикации. Результатом является блог-подобный список последних статей на сайте.

Посредством административных инструментов Drupal управления содержимым можно создавать другие «простые» типы узлов. Многие администраторы создают типы «news» (новости) или «announcement» (объявление) – для публикации официальных объявлений, тогда как другие пользователи могут публиковать узлы, содержащие статьи. Однако как быть, если вам потребуется хранить больше информации об узле, чем его заголовок и тело? Дополнительные модули могут привносить в систему содержимого Drupal новые виды узлов, обладающих более широкими возможностями. Одним из таких модулей (поставляемым в составе Drupal) является модуль «Poll». Когда пользователь создает новый узел типа «Poll», вместо обычного «тела содержимого» он создает список вопросов для голосования. Узлы типа «Poll» при отображении выглядят как форма для голосования и автоматически отображают число голосов по каждому пункту.

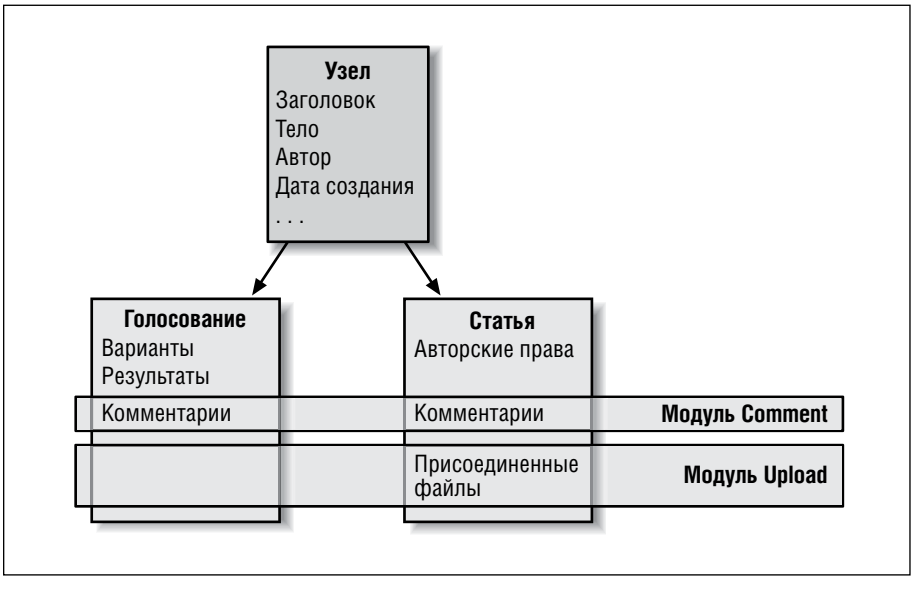


Рис. 1.6. *Все узлы в системе обладают набором общих базовых свойств; узлы могут определять дополнительные поля, а модули могут придавать узлам дополнительные функциональные возможности*

Кроме того, другие модули могут реализовывать новые свойства узлов, такие как комментарии, рейтинги, поля выгрузки файлов и другие. Из панели управления вы можете определить, узлы каких типов будут обладать этими особенностями. Эта концепция иллюстрируется на рис. 1.6.

Возможность добавления новых свойств с помощью новых модулей поверх существующей системы узлов означает, что все виды содержимого в системе Drupal относятся к одной базовой платформе, которая является одной из самых сильных сторон Drupal. Такие возможности, как поиск, рейтинг и комментарии, являются готовыми к использованию компонентами в любых новых типах узлов, которые вы только сможете определить, потому что за кулисами система Drupal всегда знает, как обращаться с базовыми элементами этих компонентов – узлами.

Использование дополнительных модулей для добавления новых типов узлов или для добавления новых полей к существующим типам является распространенной задачей в Drupal. Продвигаясь по книге, мы рассмотрим некоторые из имеющихся сотен дополнительных модулей, и вы узнаете, как создавать сложные типы содержимого, используя базовые инструменты.

Способы организации содержимого

Еще одним важным строительным блоком является весь комплекс инструментов и приемов организации узлов, составляющих информаци-

онное наполнение сайта. Веб-сайты первого поколения группировали страницы с помощью папок и каталогов. Сайты второго поколения для отображения содержимого различных типов использовали отдельные сценарии. В свою очередь, система Drupal практически все содержимое представляет как набор узлов. Как можно разбить сайт на отдельные разделы по темам, на блоги конкретных пользователей или реализовать некоторую иную организационную схему?

Во-первых, каждый отдельный узел на сайте получает свой собственный адрес URL. По умолчанию этот адрес URL выглядит примерно так: *http://www.example.com/node/1*. Эти адреса URL можно преобразовать в более дружественный для пользователя вид, например: *http://www.example.com/about*, с помощью модуля Path, встроенного в систему Drupal. В организационных целях все эти узлы интерпретируются как единый «пул» содержимого. Любые другие страницы на сайте – тематические обзоры, последние новости и другие – создаются посредством извлечения списков узлов, соответствующих определенным критериям, и отображаются различными способами. Ниже приводится несколько примеров:

Главная страница

По умолчанию главная страница сайта, построенного на базе Drupal, представляет собой обзор 10 самых последних статей. Для его создания Drupal отыскивает в пуле содержимого узлы с флагами «Published» (Опубликовано) и «Promote to front page» (Представить на главной странице), установленными в значение `true`. Кроме того, этот список сортируется так, чтобы узлы с флагом «Sticky» (Закреплен) всегда находились на самом верху; эта возможность особенно удобна для публикации горячих новостей или важных объявлений, которые должны быть видны каждому пользователю.

Модуль Taxonomy

Выше упоминалось, что модули могут добавлять к узлам новые элементы информации; именно это и делает модуль Taxonomy. Он позволяет администратору сайта определять категории тем, с которыми могут быть связаны узлы при их создании, а также ключевые слова для облака тегов в стиле блогов. Этот модуль можно использовать для создания предопределенного набора «Regions» (Регионы), представленного в виде поля в новостях или поля «Tags» (Теги) для блогеров, которое они будут заполнять вручную при отправке сообщений. Все это в терминах модуля Taxonomy называется «рубриками» и позволяет отнести страницу к той или иной рубрике на сайте. Когда посетитель просматривает одну из таких страниц, система Drupal извлекает список всех узлов, отнесенных к этой рубрике.

Модуль Blog

Модуль Blog, встроенный в систему Drupal, реализует многопользовательскую систему блогов, делая всего три вещи. Во-первых, он добавляет новый тип узлов, который называется «Blog post» (Сообщение в блоге). Во-вторых, с адресом *http://www.example.com/blog* он ассоциирует страницу со списком, в котором отображаются узлы типа «Blog» (Блог), для которых флаг «Published» (Опубликовано) установлен в значение `true`. (Если сообщение в блоге имеет флаг «Published to front page» (Публиковать на главной странице) со значением `true`, оно будет также отображаться на главной странице; система Drupal никогда не скрывает содержимое одной страницы только потому, что оно отображается на другой.) В-третьих, для каждого пользователя сайта он предоставляет отдельную страницу, где отображаются сообщения блога, принадлежащие только этому пользователю. Например, на странице *http://www.example.com/blog/1* будут отображаться все узлы с записями в блоге, которые были созданы и опубликованы пользователем 1, то есть администратором.

В составе системы Drupal поставляется ряд других модулей, которые обеспечивают возможность организации узлов различными способами, а кроме того, существуют сотни дополнительных, доступных для загрузки, модулей, позволяющие организовать сайт разными способами. Важно запомнить, что практически все «страницы» в Drupal представляют собой одну из двух вещей: определенный узел содержимого или список узлов, обладающих определенным общим набором свойств.

Типы содержимого поддержки

Помимо содержимого и списков содержимого существуют также различные способы дополнения содержимого страницы. Двумя такими способами поддержки содержимого, включенными в ядро Drupal, являются комментарии и блоки.

Комментарии – это обычные отклики пользователей на некоторые фрагменты содержимого, они существуют только в привязке к этому содержимому. Пользователи могут отправлять комментарии, чтобы представить свои мысли относительно темы узла, как это часто делается, когда в сообщении блога или на форуме поднимается спорная тема. Подобно узлам, хотя и в меньшей степени, комментарии могут с помощью сторонних модулей расширяться дополнительными свойствами, такими как рейтинги или выгружаемые файлы.

Комментарии имеют большое число параметров настройки: комментарии могут отображаться в виде древовидной структуры или в виде плоского списка; комментарии могут сортироваться по дате и времени создания как в прямом, так и в обратном порядке; анонимным пользова-

телям может быть предоставлена возможность оставлять комментарии, причем в последнем случае можно сделать предоставление контактной информации обязательным.

Блоки – это виджеты, которые присутствуют в таких областях страницы, как боковые меню, нижние колонтитулы и заголовки. Обычно они используются для отображения полезных ссылок или динамических списков, таких как «Most popular content» (Наиболее читаемые статьи), «Latest comments» (Последние комментарии), и похожих элементов. Блок пользователей управляет доступностью информации для посетителей вашего сайта; узлы отвечают за отображение содержимого; а блоки помогают встроить отдельный элемент содержимого в контекст структуры вашего сайта.

В большинстве случаев блоки отображают различное содержимое для разных зарегистрировавшихся пользователей: например, блок «Comments by your buddies» (Комментарии ваших друзей) будет отображать список сообщений, оставленных теми пользователями, которых текущий пользователь добавил в список своих друзей. Обычно все пользователи, выполнившие вход, видят различные списки. Кроме того, блоки можно настроить так, что они будут появляться только на определенных страницах или, наоборот, будут скрыты только на определенных страницах.

Получение справки

Легко полагаться на ту функциональность, которую можно получить бесплатно, используя открытое программное обеспечение. Но не стоит забывать, что само сообщество пользователей Drupal может стать важным строительным блоком вашего веб-сайта!

По мере изучения практических примеров в этой книге вы можете столкнуться с некоторыми проблемами, характерными для вашего окружения. Проблемы могут также возникать при переходе на использование новых версий модулей. К счастью, у сообщества Drupal имеется богатый выбор ресурсов, где можно получить помощь в исследовании самых противных ошибок, с которыми вы только можете столкнуться:

- Руководства Drupal, расположенные по адресу *http://drupal.org/handbooks*[1], содержат массу информации обо всем, начиная от философии сообщества и заканчивая подробными сведениями, касающимися разработки Drupal.
- Руководство для начинающих, по адресу: *http://drupal.org/gettingstarted*, содержит некоторую информацию, которая будет вам особенно полезна в первые два часа знакомства с Drupal.

[1] Существует русскоязычный сайт Drupal.ru, где имеется похожий раздел *http://drupal.ru/book*. – *Примеч. перев.*

- Сборник вопросов и ответов по устранению проблем – по адресу: *http://drupal.org/Troubleshooting-FAQ* содержит полезные советы и рекомендации с расшифровкой сообщений об ошибках, с которыми вы можете столкнуться.
- За персональной помощью по любому вопросу, от подготовки к установке до проблем, связанных с обновлением, можно попробовать обратиться на форум «Support» (Поддержка), по адресу: *http://drupal.org/forum/18*[1].
- Если ваш вопрос касается определенного модуля, можно послать «запрос на поддержку» (или «отчет об ошибке», если это явная проблема) в очередь вопросов, который будет передан лицу, сопровождающему модуль. Полезный видеоурок о том, как использовать очередь вопросов на сайте Drupal.org, находится по адресу *http://drupal.org/node/273658*. Кроме того, очередь вопросов обсуждается также в Приложении B.
- На сайте *irc.freenode.net* имеется канал IRC #drupal-support для тех, кто предпочитает общаться в диалоговом режиме.

В отличие от #drupal-support канал #drupal на сайте *irc.freenode.net не* является каналом поддержки. Этот канал служит местом сбора разработчиков, где они предоставляют помощь друг другу, обмениваются информацией и обсуждают вопросы дальнейшего развития проекта Drupal в целом. Старайтесь участвовать в этих обсуждениях, чтобы быть в гуще событий сообщества, и задавайте свои вопросы, имеющие отношение к программированию, но помните, что такие вопросы, как «Какой параметр нужно включить, чтобы сделать то-то и то-то?» и «Какой модуль следует использовать для таких-то целей?» может вывести людей из равновесия.

При составлении своего вопроса всегда желательно предварительно провести самостоятельные исследования и только потом задавать прямые и недвусмысленные вопросы. Вопрос: «Модуль Foo дает мне ошибку “Invalid input” (Неверный ввод), когда я пытаюсь отправить слово “Steve” в качестве имени поля. Я попытался найти существующие решения и обнаружил сообщение *http://drupal.org/node/1234*, описывающее эту проблему, но приведенное там решение не устранило мою ошибку. Мог бы кто-нибудь дать мне какие-либо указания?», – выглядит намного лучше, а ответ на него придет гораздо быстрее и будет более содержательным, чем если бы вы задали вопрос: «Почему модуль Foo не работает? От вас, разработчиков, мало проку!» или: «Как мне построить сайт на базе Drupal?». Нередко в процессе подготовки вопроса к отправке при вводе подробного описания проблемы для того, что-

[1] Аналогичная ветка форума на русскоязычном ресурсе Drupal.ru: *http://drupal.ru/forum/support*. – *Примеч. перев.*

бы кто-то другой смог разобраться в ней, вы сможете самостоятельно найти решение!

В заключение

В этой главе вы узнали, что такое Drupal. Вы познакомились с историей веб-сайтов и систем управления содержимым, чтобы лучше понять проблемы, связанные с поддержкой растущего сайта. Мы исследовали концептуальные строительные блоки, используемые системой Drupal при построении сайтов нового поколения, и то, как они соотносятся друг с другом. Мы также познакомились со множеством способов получения помощи в случае появления проблем. В следующей главе мы объединим все это вместе и создадим наш первый веб-сайт на базе Drupal!

2

Запускаем Drupal

Эта глава предназначена для тех, кто только начинает знакомство с Drupal. В ней дается обзор возможностей системы, а также вводятся определения иногда не совсем понятных терминов и демонстрируется, как можно с помощью Drupal построить простой веб-сайт. Читатели, уже знакомые с Drupal, также могут просмотреть эту главу, так как последующие главы опираются на сведения, которые приводятся здесь. К концу главы вы будете знать, как выполнять в Drupal такие задачи администрирования, как настройка модулей, работа с типами содержимого и настройка системы навигации по сайту.

В этой главе предполагается, что вы уже установили и запустили Drupal. За помощью по этому вопросу обращайтесь к приложению А, а также к руководству для начинающих по адресу: *http://drupal.org/getting-started*.

В этой главе будут представлены следующие модули:

Node (базовый)

Позволяет добавлять новое содержимое и создавать свои типы содержимого.

Comment (базовый)

Дает пользователям возможность оставлять свои комментарии к элементу содержимого.

User (базовый)

Обеспечивает возможность регистрации пользователей, а также реализует надежные системы ролей и прав доступа

Block (базовый)

Добавляет динамические боковые врезки и другое дополнительное содержимое.

Menu (базовый)

Обслуживает систему навигации по веб-сайту на базе Drupal.

Path (базовый)

Обеспечивает поддержку дружественных адресов URL, таких как *http://www.example.com/about*, вместо *http://www.example.com/node/1*.

Administration Menu (http://drupal.org/project/admin_menu)

Реализует динамические раскрывающиеся меню для ускорения доступа к выполнению административных задач.

Contact (базовый)

Простая форма, которая может использоваться посетителями сайта для отправки своих вопросов владельцам веб-сайта.

Blog (базовый)

Реализует простой и быстрый многопользовательский блог

Taxonomy (базовый)

Мощная система классификации.

Filter (базовый)

Важный и часто недооцениваемый модуль, который является ключевым в системе безопасности Drupal.

FCKeditor (http://drupal.org/project/fckeditor)

Визуальный редактор («What You See Is What You Get», WYSIWYG – что вижу, то и получаю), который позволяет тем, кто не знаком с языком разметки HTML, создавать содержимое веб-сайта, применяя элементы форматирования.

IMCE (http://drupal.org/project/imce)

Дополнительный модуль, который может работать с такими редакторами, как FCKeditor, и позволяет легко добавлять изображения в содержимое веб-сайта.

Построенный веб-сайт будет выглядеть, как показано на рис. 2.1 и по адресу: *http://jumpstart.usingdrupal.com*.

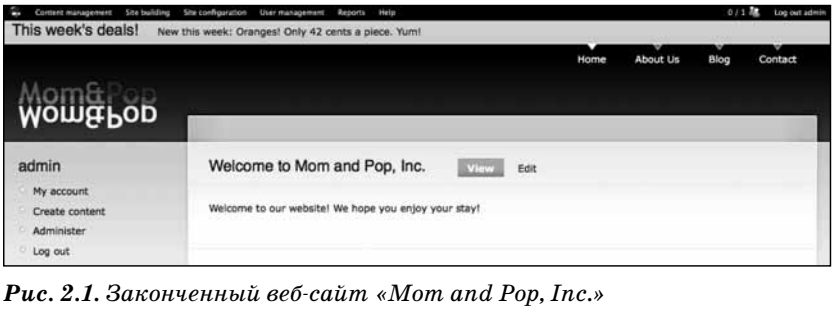


Рис. 2.1. *Законченный веб-сайт «Mom and Pop, Inc.»*

Задача

Компания «Mom and Pop, Inc.» – это небольшой продуктовый магазин на Среднем Западе США, которым совместно управляют Дженни и Майк. В настоящее время их присутствие в Интернете представлено длинной статичной страницей HTML, в которой приводится общая информация, такая как происхождение компании, часы работы, местоположение и рекламные мероприятия, проводимые в настоящее время.

Так как ни Майк, ни Дженни не сильны в редактировании кода страницы вручную, то для еженедельного обновления содержимого страницы они наняли на эту ручную работу свою соседку Голди. Эта работа достаточно утомительна и совсем неинтересна, поэтому Голди не сразу бросается выполнять ее. В результате сайт часто содержит устаревшую информацию и не отрабатывает вложенные на его поддержание средства.

Майк и Дженни хотели бы получить новый, свежий сайт, которым они смогли бы управлять сами, заполняя поля веб-форм вместо того, чтобы вручную править код разметки страницы. Требуется, чтобы сайт содержал статические страницы, такие как «Home» (Главная) и «About» (О компании), а кроме того, страницу, где отображалась бы информация о еженедельных рекламных акциях. Они также хотели бы иметь форму для контактов, чтобы получать заказы клиентов.

Майк и Дженни также хотели бы получить блог, в котором можно было бы обмениваться информацией о различных событиях в жизни магазина или местных событиях. Посетителям сайта требуется предоставить возможность комментировать сообщения в блоге, при этом комментарии анонимных посетителей должны пройти одобрение, прежде чем быть опубликованными.

Ни Майк, ни Дженни не занимаются программированием, поэтому важно, чтобы сайт позволял легко редактировать содержимое даже тому, кто не обладает знаниями HTML. И, наконец, у сайта должны быть эмблема, логотип и фирменные цвета, чтобы он имел «свое лицо».

В последнее время Голди много слышала о Drupal, поэтому она решила задействовать эту систему для своего проекта.

Примечания к реализации

В разделе «Примечания к реализации» в каждой главе будут обсуждаться, сравниваться и противопоставляться различные варианты удовлетворения потребностей клиента в системе Drupal, а также будет описываться, как авторы пришли к решению использовать те или иные возможности.

Основы

Практически все функциональные возможности, которые необходимы для удовлетворения потребностей компании «Mom and Pop, Inc.»,

реализованы в составе программного обеспечения Drupal, доступного для загрузки, которое называется «ядром» Drupal. Модуль Node имеет встроенные средства создания различных типов содержимого, включая статические страницы, которые прекрасно подойдут для страниц Home (Главная) и About (О компании). Чтобы присвоить этим страницам красивые и осмысленные адреса URL, такие как *http://www.example.com/about*, мы будем использовать базовый модуль Path.

Система Drupal также предоставляет надежную систему ролей и прав доступа, которую можно использовать для отделения задач, выполняемых Голди (сопровождение веб-сайта), от задач, выполняемых Майком и Дженни (управление содержимым веб-сайта), и от задач, выполняемых клиентами сайта (которым будет позволено только оставлять свои комментарии).

В системе Drupal имеется встроенный модуль Blog, который прекрасно подойдет для Майка и Дженни и позволит им сообщать о новостях своего магазина. А модуль Comment позволит посетителям вступать в дискуссии.

Кроме того, в составе системы Drupal имеется модуль Contact, который можно использовать для создания простой формы с контактной информацией для любого веб-сайта. Имеется возможность определить различные категории, и с каждой из них связать отправку сообщений на отдельный адрес электронной почты. Это очень удобно, например, если поддержкой различных отделов занимаются разные люди.

Простота редактирования и обработки изображений

Без каких-то ухищрений все содержимое в системе Drupal вводится в формате HTML, при этом с помощью встроенного модуля Filter обеспечиваются некоторые меры безопасности. С ручным вводом тегов HTML прекрасно справляются обычные веб-мастера, хорошо ориентирующиеся в HTML, однако большинство «обычных» пользователей (особенно те, кто не имеет отношения к вычислительной технике), как правило, не горят желанием вводить код странного вида, чтобы, например, создать простой список.

Кроме того, вполне естественно, что Майк и Дженни помимо текстового содержимого захотят поместить на сайте изображения: магазина, купонов рекламной акции, проводимой на текущей неделе, или фотографии своих детей. Ядро Drupal не имеет встроенных средств для работы с изображениями, – тогда как можно решить эту проблему?

Решить обе эти проблемы можно с помощью богатой библиотеки дополнительных модулей. Ниже в этой главе, в разделе «В центре внимания: редактирование содержимого», мы подробнее обсудим имеющиеся решения. Но сначала обсудим базовые понятия.

В центре внимания: управление содержимым

Основная функция системы Drupal состоит в том, чтобы дать возможность администраторам веб-сайтов управлять их содержимым. В этом разделе мы рассмотрим некоторые основные инструменты управления содержимым в Drupal.

Содержимое

Как уже говорилось в главе 1, каждый элемент содержимого в системе Drupal – от статической страницы до блога или голосования – называется узлом. В составе Drupal по умолчанию поставляется два типа содержимого: тип Page (Страница), предназначенный для определения узлов со статическим содержимым, таким как страница «About Us» (О компании), и тип Story (Статья), предназначенный для определения узлов, которые будут создаваться достаточно часто, например новости. Как и все остальное в Drupal, типы содержимого допускают возможность полной перенастройки. На рис. 2.2 показана страница Create content (Создать материал)[1] для типичного сайта на базе Drupal с несколькими доступными типами содержимого. Эту страницу можно найти в разделе *node/add* (*http://www.example.com/node/add*).

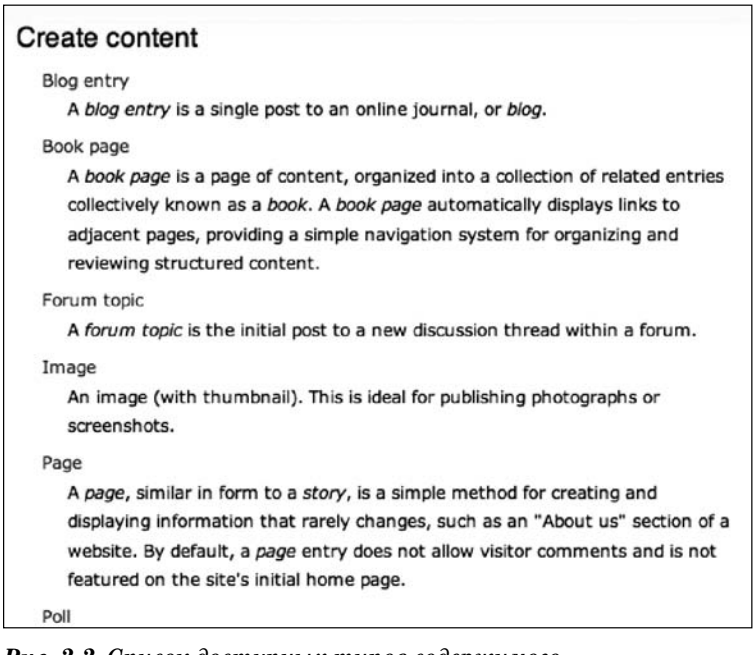


Рис. 2.2. *Список доступных типов содержимого*

[1] Здесь и далее перевод элементов интерфейса будет приводиться в соответствии с русскоязычной версией Drupal. – *Примеч. перев.*

Как упоминалось во вступлении, в последующем для обозначения переходов в пределах сайта будет использоваться сокращенная форма записи, например Create content (Создать материал) (*node/add*).

На рис. 2.3 показан пример типичной формы, используемой для добавления или редактирования элемента содержимого. Каждый узел имеет поле Title (Заголовок), который служит для идентификации узла в списках содержимого и определяет заголовок веб-страницы. Большинство узлов имеют также поле Body (Содержимое), которое хранит основное содержимое. Даже если внушительный набор параметров внизу формы может показаться вам пугающим, – не волнуйтесь. Обычные посетители сайта, как правило, не имеют права изменять разделы Menu settings (Настройки меню), Authoring information (Информация об авторе) и другие разделы с настройками, поэтому они просто не отображаются для таких пользователей.

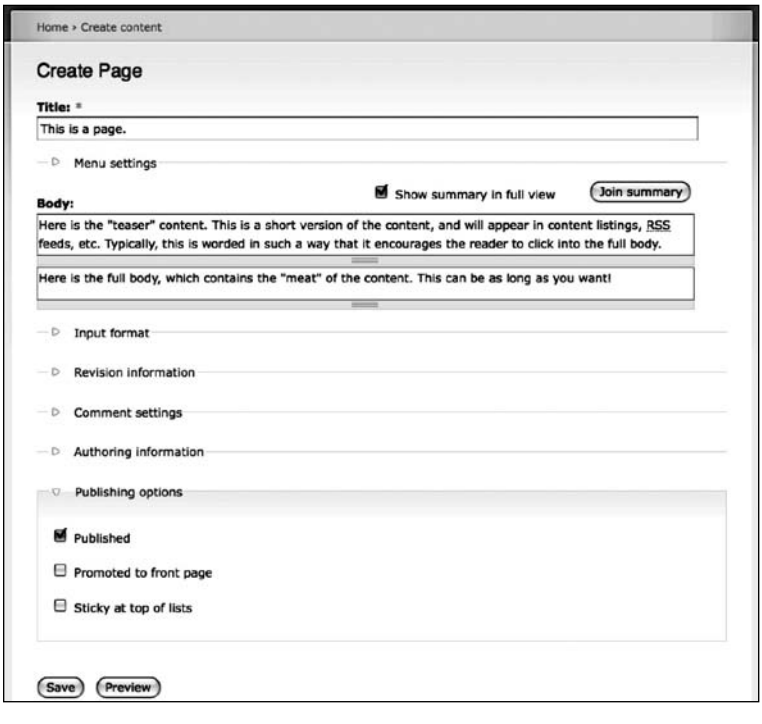


Рис. 2.3. *Типичная форма создания узла в системе Drupal*

Поле Body (Содержимое) узла может быть разбито на *полное представление*, содержащее все содержимое, и *анонс*, который является краткой аннотацией статьи, побуждающей посетителей продолжить чтение. В большинстве случаев анонсы отображаются в списках, в лентах RSS

и в других местах. Полное тело узла отображается только при непосредственном просмотре этого элемента содержимого. Вы можете указать, следует ли включать анонс в полное представление с помощью флажка Show summary in full view (Показывать анонс в полной версии).

Люди, начинающие использовать Drupal и имеющие опыт веб-разработки с помощью таких инструментов, как Dreamweaver, часто неправильно интерпретируют понятие «страницы» в Drupal. Другие инструменты веб-разработки под «страницей» подразумевают все содержимое окна броузера, от логотипа в верхнем левом углу до примечания об авторских правах в правом нижнем углу. Но в Drupal под узлом типа Page подразумевается только *содержимое* данной веб-страницы: ее заголовок, ее тело и любые другие дополнительные свойства, такие как подпись или рейтинг.

Узлы могут обладать различными параметрами, применяемыми к ним самим, включая возможность отслеживания и откат ревизий (об этом будет рассказываться в главе 5) и возможность по умолчанию быть неопубликованными, чтобы после добавления они не появлялись на сайте сразу же. Эти параметры могут устанавливаться для каждого узла в отдельности или определяться как значения по умолчанию для всех узлов данного типа в разделе Administration (Администрирование) типов содержимого: Administer (Администрирование)→Content management (Содержание)→Content types (Типы материалов) (*admin/content/types*), как показано на рис. 2.4.

Изменение значений по умолчанию для параметров настройки типов содержимого не применяется к уже созданному содержимому. Поэтому необходимо потратить некоторое время, чтобы определиться, какие значения должны иметь эти параметры настройки узлов данного типа, прежде чем приступать к заполнению сайта содержимым.

Узлы, для которых помечен флажок Promoted to front page (Помещать на главную), будут отображаться в списке на главной странице, доступном по адресу *http://www.example.com/node*, как показано на рис. 2.5. Узлы отображаются один за другим, при этом первыми будут следовать узлы с отмеченным флажком Sticky at top of lists (Закреплять вверху списков), а остальная часть списка будет упорядочена в хронологическом порядке, начиная с самых свежих.

Внесение изменений в содержимое можно выполнить непосредственно в самом узле сразу же после его создания, или на странице Administer (Администрирование)→Content management (Содержание)→Content (Содержимое) (*admin/content/node*), как показано на рис. 2.6, где можно удалять содержимое, публиковать или снимать с публикации, а также выполнять различные операции.

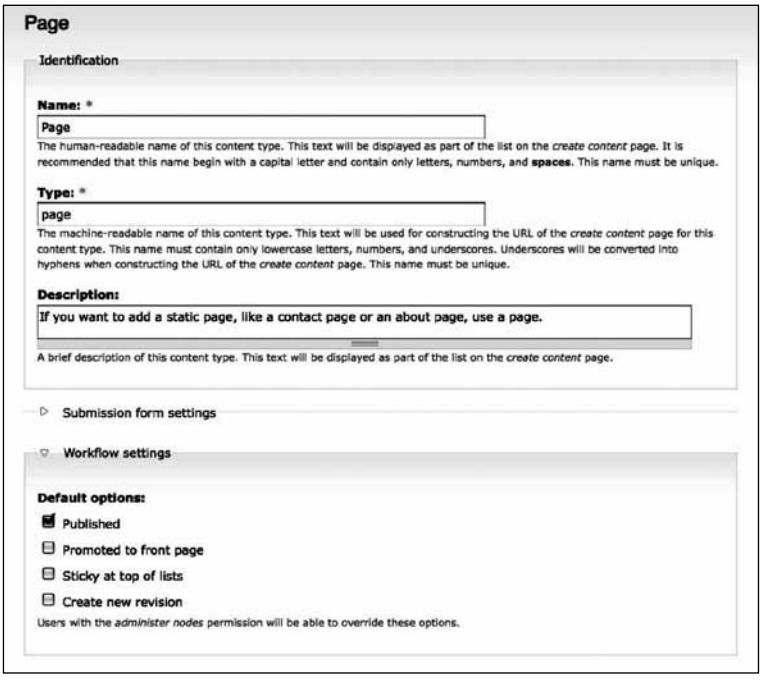


Рис. 2.4. *Форма администрирования типов содержимого*

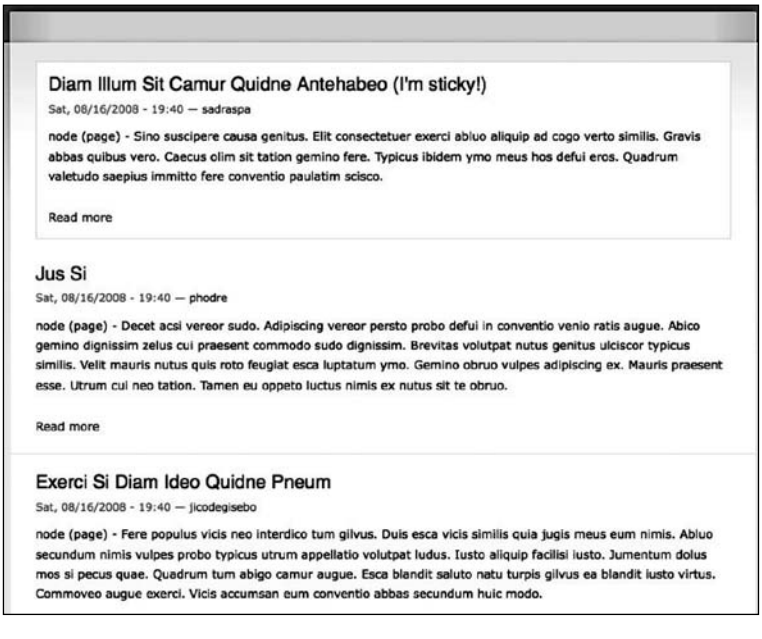


Рис. 2.5. *Вид главной страницы по умолчанию*

Название «главная страница» не совсем удачное. Список, находящийся по адресу */node*, – это главная страница только по умолчанию – вы можете назначить домашней любую другую страницу, что мы и сделаем ниже, в этой же главе, в разделе «Практика: управление содержимым».

Даже при том, что такое представление содержимого по умолчанию является наиболее типичным, тем не менее, вы можете создать список, содержащий узлы практически любого типа, задействовав модуль Views (*http://drupal.org/project/views*), который будет подробнее рассматриваться в следующей главе и широко использоваться на протяжении всей книги.

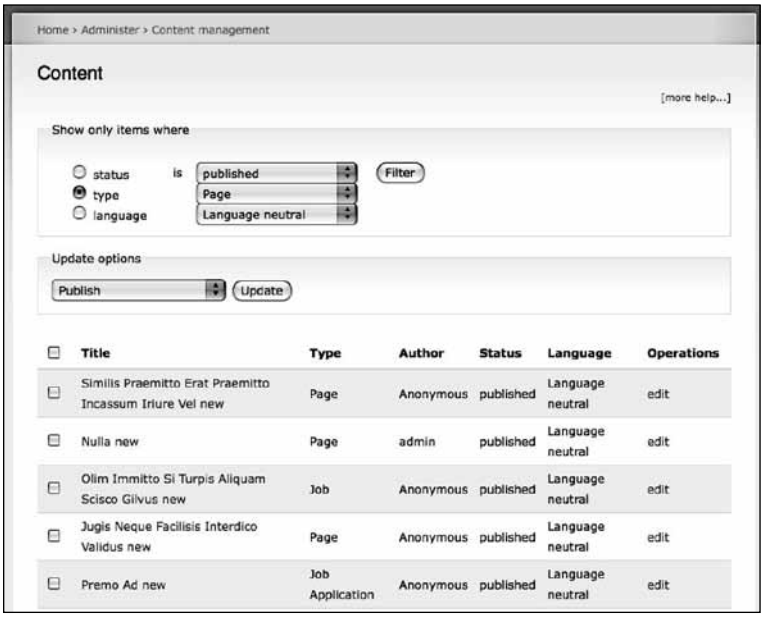


Рис. 2.6. *Страница администрирования содержимого*

Комментарии

Базовый модуль Comment позволяет посетителям сайта оставлять свои реплики по поводу содержимого узла, что дает возможность непосредственно в теме вступать в обсуждение с автором и с другими посетителями. На рис. 2.7 показаны комментарии в действии.

Большинство типов содержимого по умолчанию позволяют оставлять комментарии, хотя для содержимого типа Page эта возможность по умолчанию отключена (так как нет смысла обсуждать, например, страницу «About Us» (о компании)). По адресу Administration (Администрирование) типов содержимого Administer (Администрирование)→Content

management (Содержание)→Content types (Типы материалов) (*admin/content/types*) вы найдете настоящий «шведский стол» параметров настройки комментариев для каждого типа содержимого, которые управляют самыми разными аспектами, начиная от того, где должны отображаться комментарии, и заканчивая тем, должны ли анонимные пользователи вместе с комментарием оставлять свою контактную информацию. Некоторые из этих параметров мы рассмотрим подробнее ниже в этой главе.

При необходимости комментарии могут также отправляться в очередь для модерирования, вместо того чтобы сразу же появляться на сайте, что может использоваться, как средство борьбы со спамом.

Рис. 2.7. *Модуль Comment позволяет посетителям обсуждать содержимое узла*

Система Drupal предлагает большое число модулей для борьбы со спамом и нежелательным содержимым. Некоторые возможности мы рассмотрим ниже в этой главе, в разделе «В центре внимания: инструменты модерирования содержимого».

Навигация

Рука об руку с созданием содержимого идет предоставление возможности отыскивать это содержимое на сайте. Для этих целей система Drupal предоставляет встроенный модуль Menu. Меню содержат ссылки на различные веб-страницы на сайте Drupal. В составе Drupal поставляются меню трех видов:

Навигация

Главное системное меню. На практике это меню по умолчанию представляет собой «место свалки» ссылок, создаваемых модулями, включая ссылки для выполнения административных задач.

Основные ссылки

Пустое меню, предоставляемое для реализации собственной системы навигации, обычно отображается в самом заметном месте сайта. В основное меню обычно помещаются ссылки на такие разделы сайта, как «Home» (Главная) и «Blog» (Блог).

Дополнительные ссылки

Еще одно пустое меню, предоставляемое для нужд навигации, но отображаемое менее приметно. Как правило, в дополнительное меню помещаются ссылки на второстепенные страницы, такие как «Terms of Service» (Условия обслуживания) или «FAQ» (Часто задаваемые вопросы).

Для типов содержимого Page и Story вам *не обязательно* предварительно создавать ссылки для первичного и вторичного меню. Эта возможность – лишь отправная точка, которая поможет быстро настроить и запустить сайт.

На рис. 2.8 показан пример всех трех меню в базовой теме по умолчанию «Garland». Подробнее темы будут рассматриваться ниже в этой главе, в разделе «В центре внимания: темы оформления». Обратите также внимание, что вы легко можете изменять порядок следования пунктов меню, перетаскивая их мышью, ухватив за ярлык с изображением крестика.

В обычных условиях первичное и вторичное меню – это два разных меню, однако по адресу Administer (Администрирование)→Site building (Конструкция сайтов)→Menus (Меню)→Settings (Настройки) (*admin/build/menu/settings*) имеются параметры настройки, с помощью которых можно создать общее меню, содержащее ссылки из первичного и вторичного меню. При этом создается эффект «раскрывающегося меню», в котором ссылки первичного меню отображаются как элементы верхнего уровня, а ссылки вторичного меню отображаются как подпункты, после щелчка на элементе верхнего уровня.

Блоки

Блоки – это небольшие элементы содержимого, которые можно помещать в страницы. Примерами некоторых блоков, используемых по умолчанию, являются: блок «Who's online» (Кто сейчас на сайте), где отображается список пользователей, присутствующих в настоящий момент на сайте; блок «User login» (Вход), который отображает форму

ввода имени пользователя и пароля; и «Recent comments» (Последние комментарии), который отображает список последних комментариев, оставленных на сайте. Даже навигационное меню и кнопка «Powered by Drupal» (Работает под управлением Drupal) внизу страницы, как показано на рис. 2.8, являются блоками! Имеется также возможность создавать свои собственные блоки, например для отображения объявлений о предстоящих событиях.

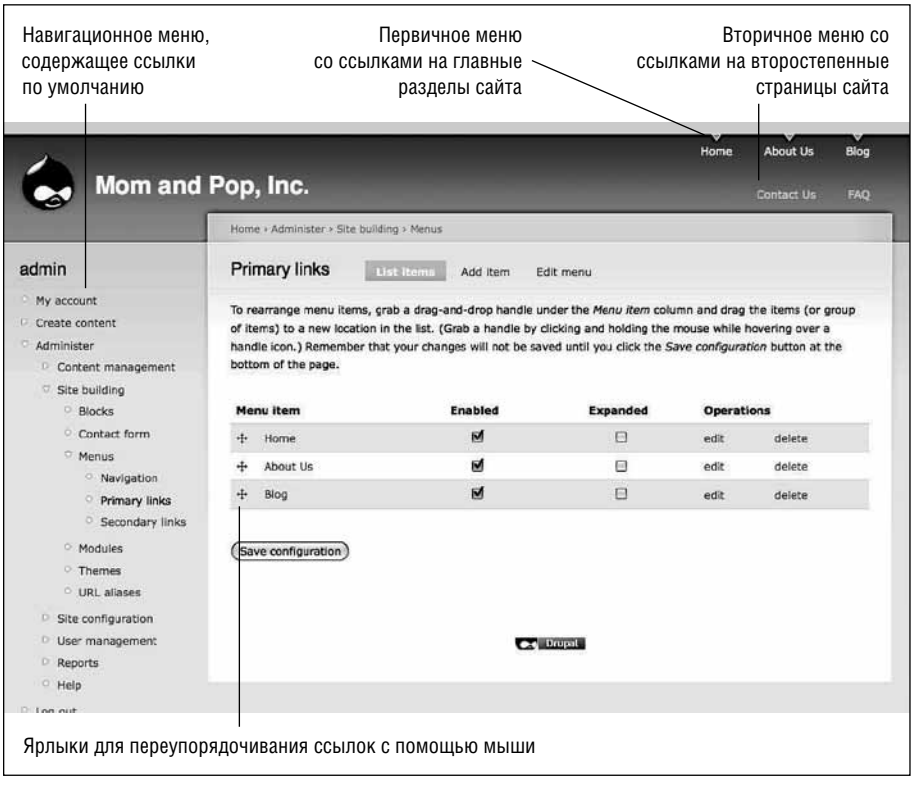


Рис. 2.8. *На странице администрирования меню показаны навигационное, первичное и вторичное меню*

На рис. 2.9 показана страница управления блоками, находящаяся по адресу Administer (Администрирование)→Site building (Конструкция сайтов)→Blocks (Блоки) (*admin/build/block*). Блоки помещаются внутри *областей* страницы. Примерами областей могут служить left sidebar (левая колонка), footer (подвал) и content (содержимое). Имена областей и их местоположение на странице могут отличаться от темы к теме – некоторые темы могут определять дополнительные области, такие как Banner ad (Рекламный баннер), удалять или изменять некоторые области по умолчанию. По этой причине блоки должны настраиваться для каждой темы отдельно. Подробнее области будут рассматриваться ниже в этой главе,

в разделе «В центре внимания: темы оформления», когда мы будем говорить о темах оформления. Как и в случае с меню, имеется возможность перетаскивать блоки из одной области в другую с помощью мыши.

Block	Region	Operations
Left sidebar		
Navigation	Left sidebar	configure
User login	Left sidebar	configure
Right sidebar		
Who's online	Right sidebar	configure
Content		
No blocks in this region		
Header		
No blocks in this region		
Footer		
Powered by Drupal	Footer	configure
Disabled		
Primary links	<none>	configure
Recent comments	<none>	configure
Secondary links	<none>	configure

Рис. 2.9. *Страница администрирования блоков*

Будьте внимательны, используя и предоставляя доступ к интерпретатору PHP на сайте Drupal. Несмотря на то, что это чрезвычайно мощный инструмент, возможность работать с PHP из веб-приложений, таких как Drupal, открывает брешь в системе безопасности сайта и может приводить к нарушениям в его работе. Мы еще раз напомним об этом ниже, в разделе «Практика: настройка разрешений», когда будем говорить о правах доступа.

Здесь также можно управлять видимостью блоков: например, отображать блоки только на определенных страницах или только для пользователей с определенными ролями. Можно также при желании использовать язык PHP, чтобы определить более сложные алгоритмы управления видимостью, например можно реализовать отображение бло-

ка «Help» (Справка) для всех пользователей, чей стаж составляет менее одной недели. Имеется также возможность разрешить пользователям самим управлять видимостью определенных блоков, чтобы они могли управлять представлением страницы.

Очень часто начинающие пользователи задают вопрос, чем отличаются блоки от узлов, если оба элемента предназначены для отображения содержимого. Главное правило: блоки обычно содержат информацию, дополняющую фактическое содержимое страницы. Кроме того, либо содержимое блоков постоянно изменяется (как, например, в блоке «Who's online» (Кто сейчас на сайте)), либо они содержат временную информацию, например рекламное объявление, отображаемое на главной странице в течение нескольких дней. Содержимое блока недоступно для функции поиска, поэтому, если необходимо создать постоянную ссылку на содержимое блока, лучше использовать узел.

Практика: управление содержимым

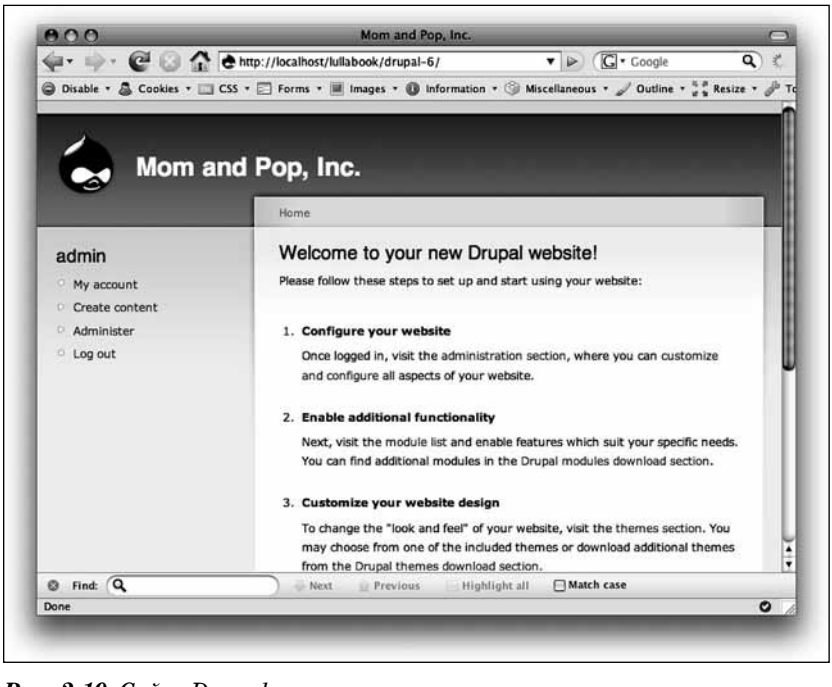


Рис. 2.10. *Сайт Drupal сразу после установки*

Наш свежесозданный замечательный сайт на базе Drupal, приведенный на рис. 2.10, выглядит довольно пустым. Добавление некоторой информации о компании «Mom and Pop, Inc.» чудесным образом пре-

образит его, сделав похожим на настоящий веб-сайт. В этом разделе мы создадим пару простых страниц, страницу «About Us» (О компании) и страницу «Home» (Главная), после чего приступим к созданию системы навигации по сайту. В качестве дополнительного оформления мы также добавим несколько блоков.

Создание содержимого

1. Сначала создадим страницу «Welcome» (Добро пожаловать), которую будем использовать в качестве главной страницы сайта. Перейдите на страницу Create content (Создать материал)→Page (Страница) (*node/add/page*).
2. Введите настройки, представленные в табл. 2.1, ниже и на рис. 2.11. Так как это статическая страница, мы также поместим в нее наше первичное меню, используя настройки, предоставляемые модулем Menu.

Таблица 2.1. Параметры настройки главной страницы

Параметр	**Значение**
Title (Заголовок)	Welcome to Mom and Pop, Inc. (Вас приветствует компания Mom and Pop, Inc.)
Параметры меню	
• Menu link title (Заголовок ссылки меню)	Home (Главная)
• Parent item (Родительский пункт)	<Primary links> (<Основные ссылки>) (по умолчанию)
• Weight (Вес)	0 (по умолчанию)
Body (Содержание)	Welcome to our website! We hope you enjoy your stay! (Добро пожаловать на наш веб-сайт! Мы надеемся, вам у нас понравится!)

3. Щелкните на кнопке Save (Сохранить), чтобы создать страницу. После создания страницы обратите внимание на путь в поле ввода адреса в вашем броузере. Он должен выглядеть, как *http://example.com/node/1*. Выпишите часть адреса, которая следует за *http://www.example.com/*, это должен быть путь *node/1*. Это идентификатор узла нашей страницы, он потребуется нам позже.
4. Затем, выполнив те же шаги, создадим страницу «About Us» (О компании). Перейдите на страницу Create content (Создать материал)→Page (Страница) (*node/add/page*).

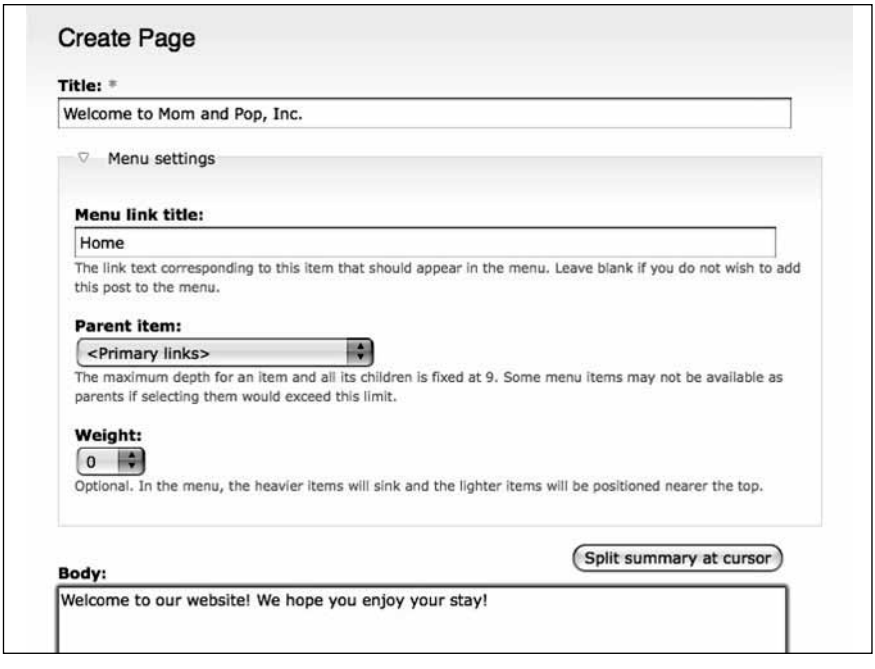


Рис. 2.11. *Создание главной страницы сайта*

5. Как и в случае создания любого другого узла, введите текст заголовка и тела. Установите настройки, как показано в табл. 2.2, и щелкните на кнопке Save (Сохранить).

Таблица 2.2. Параметры настройки страницы «About Us» (о компании)

Параметр	**Значение**
Title (Заголовок)	About Us (О компании)
Параметры меню	
• Menu link title (Заголовок ссылки меню)	About Us (О компании)
• Parent item (Родительский пункт)	<Primary links> (<Основные ссылки>) (по умолчанию)
• Weight (Вес)	0 (по умолчанию)
Body (Содержимое)	Our store has been providing organic food to the community since 1978. Come and see us at (Наш магазин продает натуральные продукты питания, начиная с 1978 года. Приходите к нам по адресу):

	123 Main Street Home Town, MN Store hours (Часы работы): 12pm–12am

6. По окончании перед вами должна появиться новая страница. Обратите внимание, что теперь в правом верхнем углу имеются два пункта меню, как показано на рис. 2.12.

Если вы забудете ввести элемент меню и покинете страницу, будет очень сложно отыскать ее снова без прямого ввода пути в адресную строку броузера, такого как *http://www.example.com/node/1*. Отыскать потерянные страницы вам поможет административная страница Content по адресу Administer (Администрирование)→Content management (Содержание)→Content (Содержимое) (*admin/content/node*).

Теперь в меню должны появиться два новых пункта, соответствующие созданным страницам. Однако если щелкнуть на заголовке сайта, в верхней части экрана, вы снова вернетесь на прежнюю страницу «Welcome to your new Drupal website» (Добро пожаловать на новый сайт Drupal), что совсем нежелательно. Нам необходимо, чтобы при входе на сайт пользователи сразу же попадали на страницу «Welcome» (Добро пожаловать). Поэтому исправим этот недостаток.

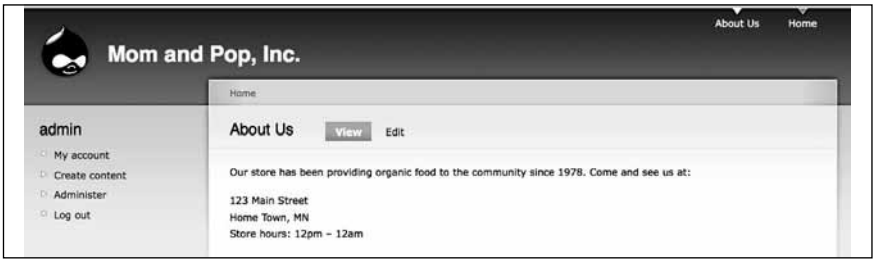


Рис. 2.12. *Окончательный вид страницы «About Us» (О компании)*

7. Перейдите на страницу Administer (Администрирование)→Site configuration (Настройка сайта)→Site information (О сайте) (*admin/settings/site-information*).
8. Прокрутите страницу вниз и найдите параметр Default front page (Главная страница по умолчанию). Введите в поле идентификатор узла, записанный ранее, когда мы создавали страницу «Home» (главная страница). В результате теперь поле должно содержать текст node/1. Щелкните на кнопке Save configuration (Сохранить конфигурацию).
9. Теперь, если снова щелкнуть на заголовке сайта, вы должны перейти на новую главную страницу, как показано на рис. 2.13.

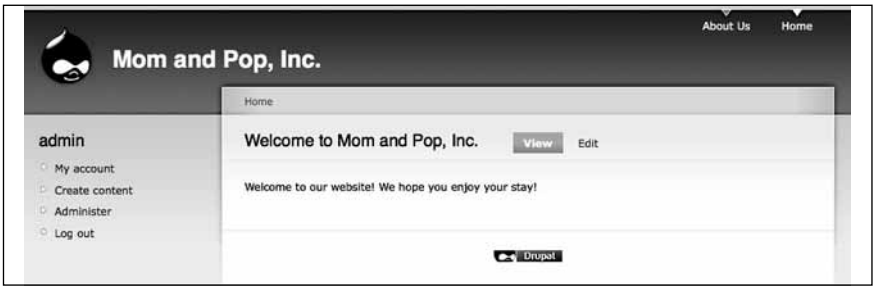


Рис. 2.13. *Окончательный вид главной страницы*

10. Нам нужно сделать еще кое-что, прежде чем двинуться дальше. Как вы уже знаете, по умолчанию в системе Drupal имеются типы содержимого Page (Страница) и Story (Статья). Однако мы не планируем публиковать статьи на веб-сайте. Если удалить неиспользуемый тип содержимого, это позволить уменьшить беспорядок и снизить вероятность ошибиться для Майка и Дженни. Перейдите на страницу администрирования типов содержимого Administer (Администрирование)→Content management (Содержание)→Content types (Типы материалов) (*admin/content/types*), изображенную на рис. 2.14.
11. Щелкните на ссылке delete (удалить) в строке для типа Story (Статья) и затем щелкните на кнопке Delete (Удалить) в диалоге, запрашивающем подтверждение.

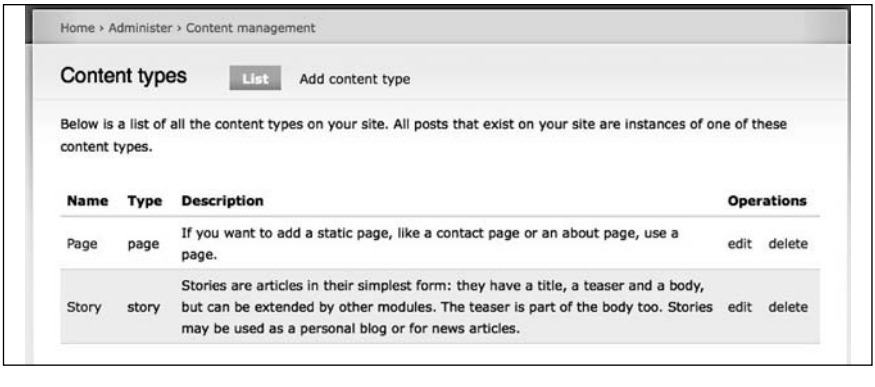


Рис. 2.14. *Страница администрирования типов содержимого*

Управление навигацией по сайту

Отлично! Теперь на нашем сайте имеется пара страниц и навигационное меню, объединяющее их. Однако можно заметить кое-что странное: пункты нашего меню, в правом верхнем углу следуют в алфавитном порядке, то есть пункт About Us (О компании) предшествует пункту Home (Главная). Было бы гораздо предпочтительнее, если бы пункт

Home (Главная) стоял на первом месте, поэтому давайте исправим этот недостаток и переупорядочим пункты меню.

1. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Menus (Меню) (*admin/build/menus*) и щелкните на ссылке Primary links (Основные ссылки) (*admin/build/menu-customize/primary-links*).
2. Используя ярлык, расположенный слева от пункта, перетащите элемент Home (Главная) так, чтобы он оказался выше элемента About Us (О компании), как показано на рис. 2.15.

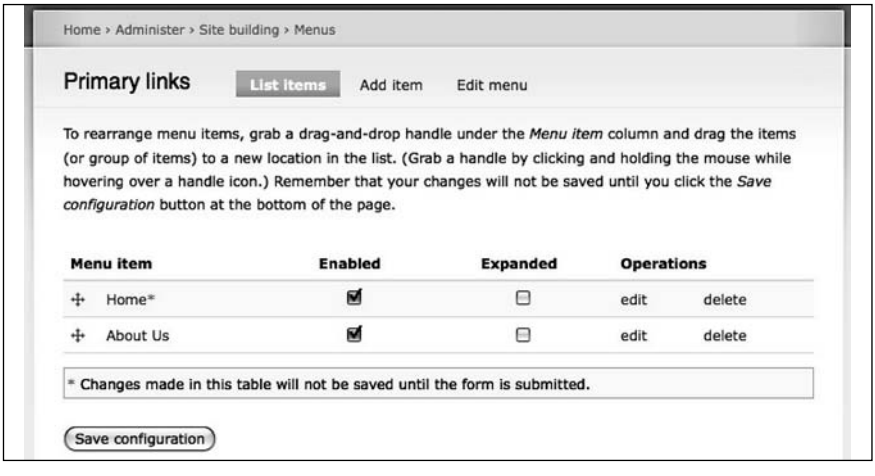


Рис. 2.15. *Страница администрирования меню*

3. Не забудьте щелкнуть на кнопке Save configuration (Сохранить конфигурацию), чтобы сохранить настройки меню.

Теперь меню должно выглядеть, как показано на рис. 2.16, с пунктом Home (Главная) впереди. Так намного лучше!

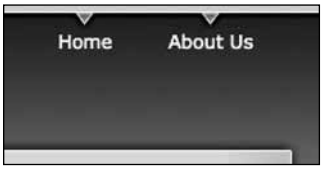


Рис. 2.16. *Переупорядоченное навигационное меню*

Настройка блоков

Теперь приступим к размещению блоков на сайте. Майк и Дженни не знают, что такое Drupal, и у них может появиться недоуменный вопрос по поводу блока «Powered by Drupal» (Работает под управлением Dru-

pal) в нижнем колонтитуле. Поэтому давайте удалим его. Кроме того, они хотят иметь возможность отображать на главной странице информацию о еженедельных рекламных акциях, что является отличной возможностью задействовать блок:

1. Перейдите на страницу администрирования блоков Administer (Администрирование)→Site building (Конструкция сайта)→Blocks (Блоки) (*admin/build/block*).
2. Для начала добавим блок с информацией о рекламной акции на текущей неделе. Для этого создадим наш собственный блок. Щелкните на вкладке Add block (Добавить блок) (*admin/build/block/add*).
3. Введите параметры настройки, как показано в табл. 2.3 и на рис. 2.17. Настройка группы параметров Page-specific visibility settings (Установки видимости для страницы) гарантирует, что блок будет отображаться только на главной странице.

Таблица 2.3. Настройки блока с информацией о рекламной акции на текущей неделе

Поле	Значение
Block-specific settings (Настройки, специфичные для блока)	
Block description (Описание блока)	Weekly deals (Каждую неделю)
Block title (Заголовок блока)	This week's deals! (На этой неделе!)
Block body (Текст блока)	New this week: Oranges! Only 42 cents each! (Новинки недели: Апельсины! Всего по 42 цента за штуку!)
User-specific visibility settings (Настройки видимости для пользователя)	
Ничего не выбрано (по умолчанию)	
Role-specific visibility settings (Настройки видимости для ролей)	
Ничего не выбрано (по умолчанию)	
Page-specific visibility settings (Установки видимости для страницы)	
Show block on specific pages (Показывать блок на определенных страницах)	Show on only the listed pages. (Отображать только на перечисленных страницах)
Pages (Страницы)	<front>

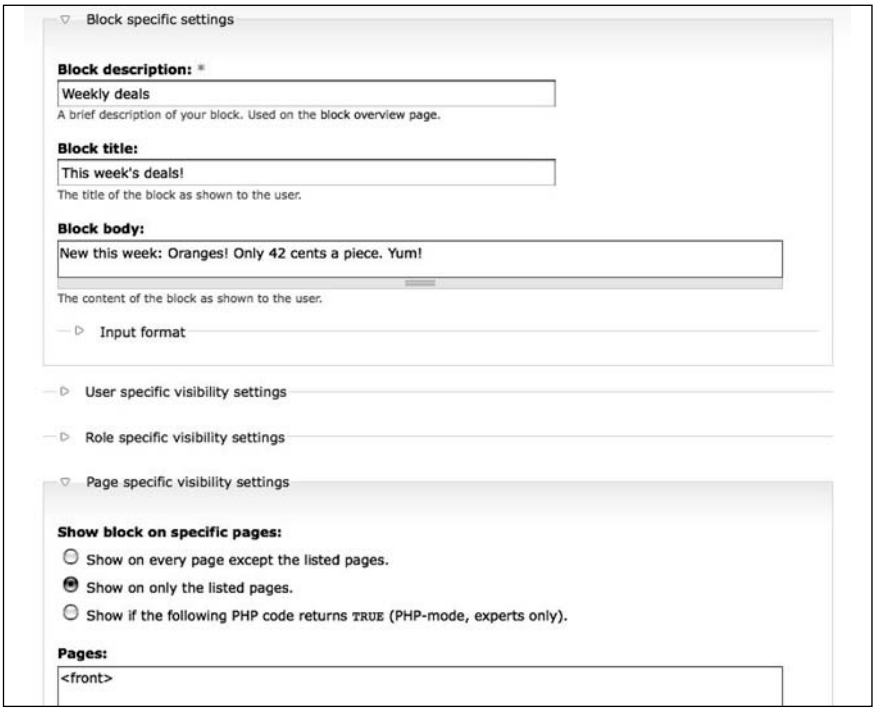


Рис. 2.17. *Форма настройки параметров блока*

4. Сохранив эту форму с помощью кнопки Save block (Сохранить блок), вы вернетесь на главную страницу администрирования блоков.
5. Теперь поместим наш новый блок на страницу. Для этого захватите указателем мыши ярлык рядом с блоком «Weekly deals» (Каждую неделю) и перетащите строку таблицы вверх, поместив ее в область Header (Заголовок). Как вы уже могли наблюдать при настройке меню, строка таблицы с блоком будет выделена желтым цветом, которым обозначается самый последний перемещавшийся элемент, а справа появится символ звездочки, показывающий, что изменения еще не были сохранены.

Перемещать блоки в таблице можно не только с помощью мыши, но и выбирая требуемую область в раскрывающемся списке, в колонке Region (Область).

6. Заодно давайте удалим блок «Powered by Drupal» (Работает под управлением Drupal). Для этого захватите указателем мыши ярлык рядом с блоком и перетащите его из области Footer (Подвал) в область Disabled (Отключено) внизу таблицы. Теперь будет выделена строка с этим блоком, а звездочки будут стоять в обоих измененных блоках. В результате страница должна выглядеть, как показано на рис. 2.18.

Block	Region	Operations
Left sidebar		
User login	Left sidebar	configure
Navigation	Left sidebar	configure
Right sidebar		
No blocks in this region		
Content		
No blocks in this region		
Header		
Weekly deals*	Header	configure delete
Footer		
No blocks in this region		
Disabled		
Powered by Drupal*	<none>	configure
Who's online	<none>	configure
Primary links	<none>	configure

Рис. 2.18. *Страница администрирования блоков после переупорядочивания блоков*

7. В заключение щелкните на кнопке Save blocks (Сохранить блоки), чтобы сохранить форму, и вернитесь на главную страницу, чтобы увидеть свои изменения в действии, как показано на рис. 2.19.

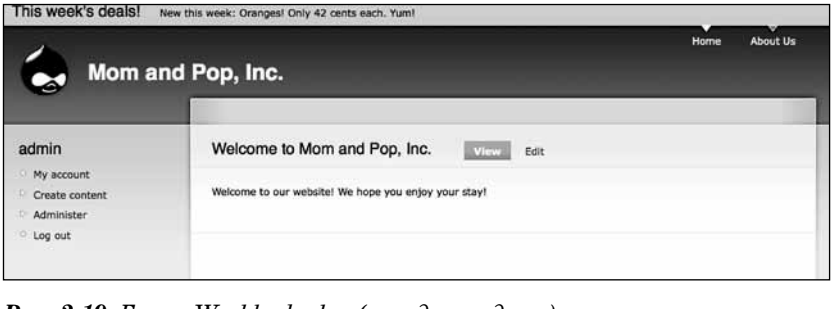


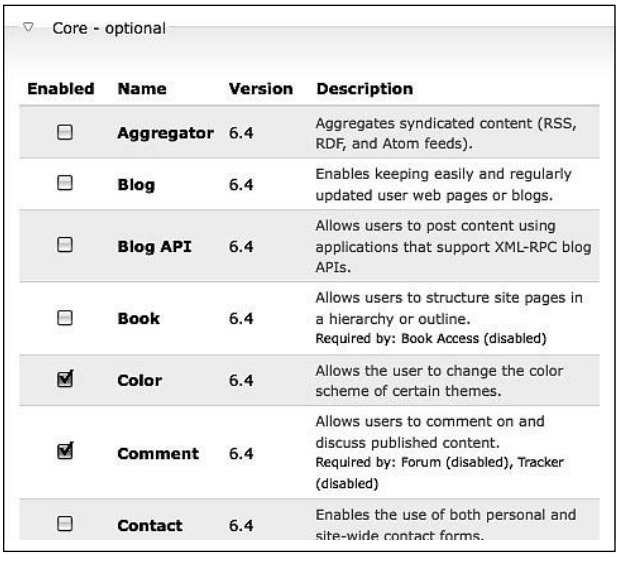
Рис. 2.19. *Блок «Weekly deals» (каждую неделю)*

В центре внимания: модули

Как уже говорилось в главе 1, модули позволяют наращивать функциональные возможности веб-сайта на базе Drupal. Всего существует два типа модулей: «базовые» модули, которые поставляются непосредственно в составе системы Drupal, и «сторонние» модули, которые бесплатно предоставляются сообществом Drupal и доступны для загрузки с сайта Drupal.org. В этом разделе мы обсудим все, что вам необходимо знать о модулях.

Страница администрирования модулей

Страница администрирования модулей доступна по адресу Administer (Администрирование)→Site building (Конструкция сайта)→Modules (Модули) (*admin/build/modules*) и изображена на рис. 2.20. На этой странице система Drupal содержит настройки функциональности веб-сайта. Взаимосвязанные модули сгруппированы вместе, внутри групп параметров, а каждая запись, соответствующая модулю, содержит его описание и сведения о версии, используемой в настоящее время. Эта информация о версии может быть чрезвычайно полезна и при выявлении проблем.



Core - optional

Enabled	Name	Version	Description
☐	**Aggregator**	6.4	Aggregates syndicated content (RSS, RDF, and Atom feeds).
☐	**Blog**	6.4	Enables keeping easily and regularly updated user web pages or blogs.
☐	**Blog API**	6.4	Allows users to post content using applications that support XML-RPC blog APIs.
☐	**Book**	6.4	Allows users to structure site pages in a hierarchy or outline. Required by: Book Access (disabled)
☑	**Color**	6.4	Allows the user to change the color scheme of certain themes.
☑	**Comment**	6.4	Allows users to comment on and discuss published content. Required by: Forum (disabled), Tracker (disabled)
☐	**Contact**	6.4	Enables the use of both personal and site-wide contact forms.

Рис. 2.20. *Страница администрирования модулей*

Модули могут подключаться и отключаться с помощью флажков в колонке Enabled (Включено), что позволяет настраивать функциональность любого сайта на базе Drupal под конкретные потребности, не перегружая без необходимости сайт.

Модуль может также иметь *зависимости*. Это означает, что для корректной работы он может требовать наличия одного или более модулей. Например, модуль Forum требует, чтобы были включены модули Comment и Taxonomy. Если вы забудете сделать это, при включении модуля Forum на экране появится диалог, запрашивающий подтверждение на включение необходимых модулей.

Кроме того, практически для каждого модуля имеется своя страница администрирования, содержащая различные параметры настройки. Чаще всего их можно найти на странице Administer (Администрирование)→Site configuration (Настройка сайта) (*admin/settings*) или Administer (Администрирование)→Site building (Конструкция сайта) (*admin/build*). Так как модули легко могут добавлять дополнительные страницы администрирования, мы рекомендуем перейти на главную страницу администратора Administration (Администрирование) (*admin*) и щелкнуть на вкладке By module (По модулям) (*admin/by-module*), как показано на рис. 2.21, чтобы увидеть все параметры настройки, разбитые на группы по модулям. Мы советуем это проделывать сразу после установки нового модуля.

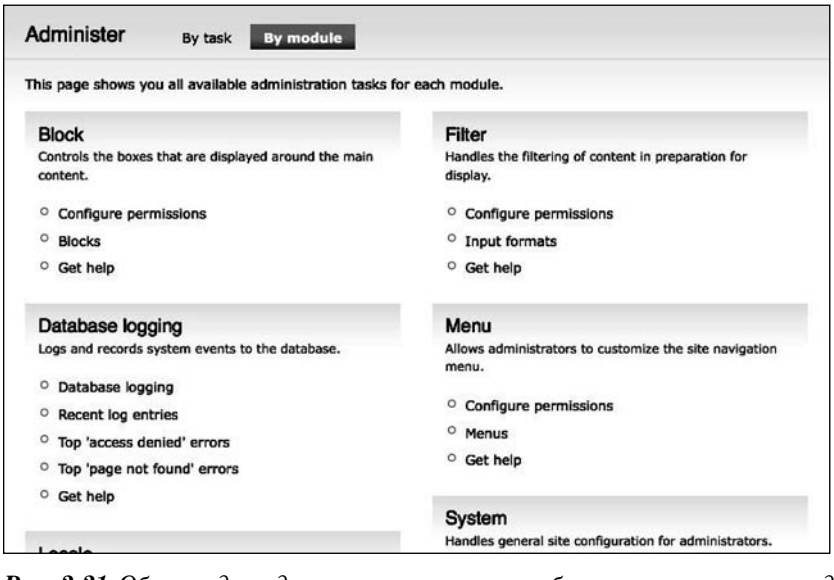


Рис. 2.21. *Обзор задач администрирования, разбитых на группы по модулям*

Поиск и установка модулей

Хотя базовые модули могут обеспечить ваш сайт основными функциональными возможностями, причем не малыми, тем не менее, истинная мощь Drupal заключена в обширном многообразии сторонних модулей, созданных сообществом. Отыскивать и загружать любые сторонние мо-

дули можно со страницы *http://drupal.org/project/Modules*, изображенной на рис. 2.22.

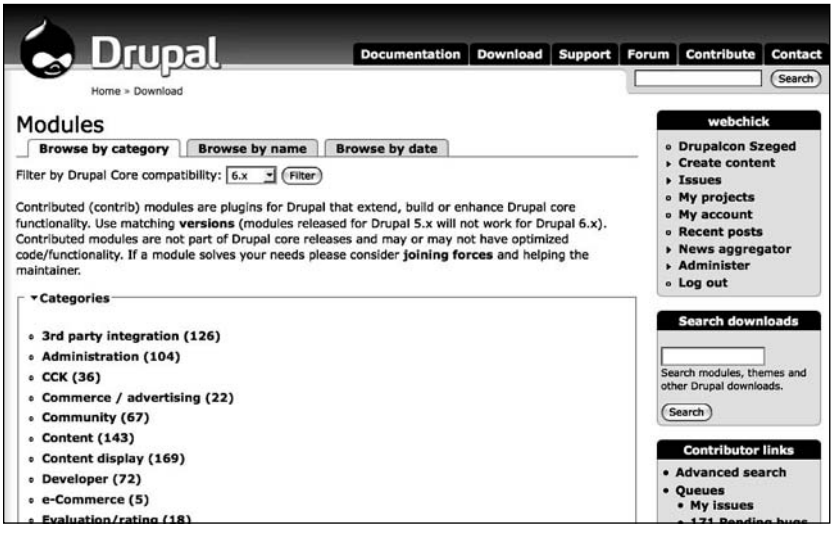


Рис. 2.22. *Страница просмотра перечня сторонних модулей на сайте Drupal.org*

Следует отметить, что модули для версии Drupal 5.x несовместимы с версией Drupal 6.x и наоборот. Поэтому очень важно выбрать в раскрывающемся списке Filter by Drupal Core compatibility (Фильтровать по совместимости с версией ядра Drupal), в верхней части страницы, используемую вами версию Drupal, чтобы в списке оставались только совместимые с ней модули. Чтобы получить список модулей, совместимых с версией Drupal 6, выберите в раскрывающемся списке пункт 6.x, а чтобы получить список модулей, совместимых с версией Drupal 5, выберите пункт 5.x. Фильтр можно использовать, только если вы зайдете на сайт Drupal.org как зарегистрированный пользователь. Учетную запись можно получить бесплатно, и она может пригодиться в некоторых ситуациях, поэтому рекомендуется создать ее.

Для каждого модуля имеется собственная страница проекта на сайте Drupal.org, как показано на рис. 2.23. Здесь вы найдете имя лидера проекта; описание; во многих случаях – снимки с экрана, демонстрирующие модуль в действии, таблицу со списком выпущенных версий, которые можно загрузить, и ссылки на другие разделы, такие как очередь запросов или документация. Вам следует загружать версию модуля, для которой написано «Recommended for 6.x» (Рекомендуется для версии 6.x) и номер которой начинается с «6.x-» (если, конечно, вы не используете Drupal 7; тогда в этом случае вам следует искать версию модуля с примечанием «Recommended for 7.x» и так далее). Дополни-

тельную информацию о соглашениях именования версий, принятых в проекте Drupal, можно найти по адресу *http://drupal.org/handbook/version-info*.

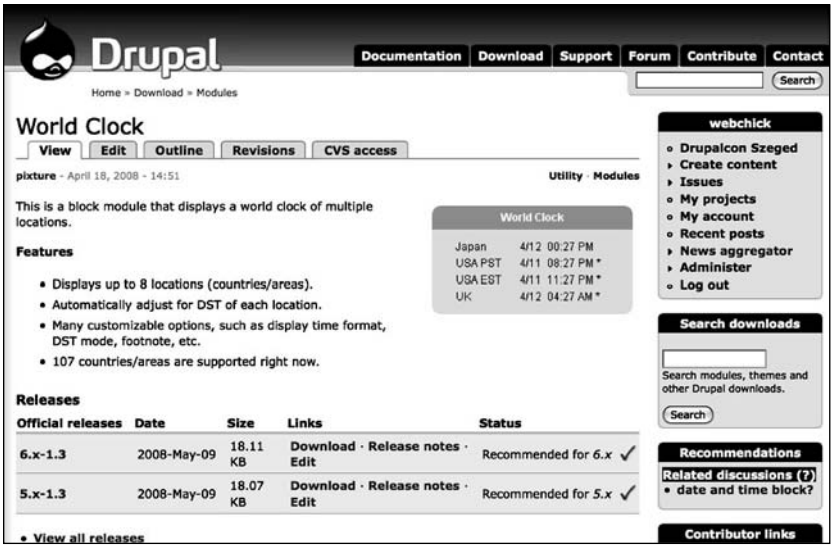


Рис. 2.23. *Страница проекта для модуля World Clock*

Один из самых сложных аспектов использования Drupal заключается в том, чтобы правильно определить, какие модули применять для решения той или иной задачи. Хотя может существовать множество модулей, решающих похожие задачи, но их качество может существенно отличаться. Приложение B целиком посвящено советам и рекомендациям по выбору правильных модулей для вашего проекта. Но недостаточно отыскать требуемый модуль – вам потребуется также обновлять его. Об обновлении модулей мы подробнее поговорим в приложении A.

После того как требуемый модуль будет найден, его следует загрузить и сохранить на локальном диске. Как и сама система Drupal, загружаемые файлы с модулями имеют расширение *.tar.gz*, это означает, что архивный файл был создан с помощью утилиты tar (такие файлы обычно называют «тарболлами») и сжат с помощью утилиты gzip. Распаковывать такие архивы и создавать каталоги с оригинальным содержимым могут многие приложения-архиваторы, такие как свободно распространяемая программа StuffIt Expander (*http://my.smithmicro.com/*) для Mac и Windows.

После распаковывания каталог с модулем нужно скопировать в каталог *sites/all/modules/*, после чего новый модуль должен появиться на странице администрирования модулей, обсуждавшейся выше.

Подробные инструкции по установке модулей можно найти в руководстве для начинающих «Getting Started» по адресу *http://drupal.org/node/258*.

Удаление модулей

Если вы решили больше не использовать какой-либо модуль, у вас есть два пути:

Отключить

Отключить модуль можно, сняв отметку на флажке в колонке Enabled (Включено) и сохранив форму. Временное отключение модуля бывает полезно при поиске неисправностей. Вы можете в любой момент повторно включить модуль, и ваш веб-сайт будет функционировать, как и прежде, так как данные для отключенного модуля не удаляются из базы данных. Отключить модуль можно только в том случае, если нет других модулей, нуждающихся в нем.

Удалить

При удалении модуль удаляется навсегда. Чтобы удалить модуль, его сначала нужно отключить, а затем отметить флажок на вкладке Uninstall (Удалить) (*admin/build/modules/uninstall*). Следует заметить, что многие, но не все, модули имеют функцию удаления.

При удалении модуля одновременно уничтожаются *все данные*, ассоциированные с этим модулем, и возможно, даже содержимое вашего веб-сайта. Поэтому нужно быть очень осторожным при использовании этой возможности и обязательно создавать резервную копию базы данных перед удалением. Обратите внимание, что при удалении модуля он не удаляется из файловой системы – вам следует выполнить эту операцию вручную.

Практика: работа с модулями

Самый простой способ разобраться с тем, как работают модули, – это попытаться установить и настроить пару модулей. Этот раздел описывает установку, включение и настройку двух модулей: базового модуля Path, встроенного в ядро Drupal, и стороннего модуля Administration Menu, который можно загрузить с сайта Drupal.org.

Модуль Path

В предыдущем разделе («Практика: управление содержимым») мы создали страницы «About Us» (О компании) и «Home» (Главная) и добавили их в меню навигации по сайту. Но если вернуться на эти страницы, можно заметить, что адрес URL страниц имеет вид *http://www.example.com/node/1*. Разве не было бы лучше, если бы эти страницы имели адреса URL более дружественные для поисковых механизмов, такие как

http://www.example.com/about? Это можно реализовать с помощью базового модуля Path.

Модуль Path позволяет задавать удобочитаемые адреса URL, называемые *синонимами URL*, которые скрывают способ определения путей в Drupal, используемый по умолчанию:

1. Для начала перейдите на страницу администрирования модулей Administer (Администрирование)→Site building (Конструкция сайта)→Modules (Модули) (*admin/build/modules*).
2. В разделе Core – optional (Ядро – дополнительные) отметьте флажок Enabled (Включено) для модуля Path, как показано на рис. 2.24, и щелкните на кнопке Save configuration (Сохранить конфигурацию) внизу формы.

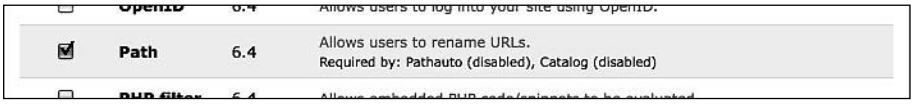


Рис. 2.24. *Включение модуля Path*

3. Щелкните на странице About Us (О компании) в меню и затем на вкладке Edit (Изменить).
4. Ближе к концу формы вы увидите новую группу параметров с подписью URL path settings (Настройки адресов), как показано на рис. 2.25. Введите в поле строку about и щелкните на кнопке Save (Сохранить).

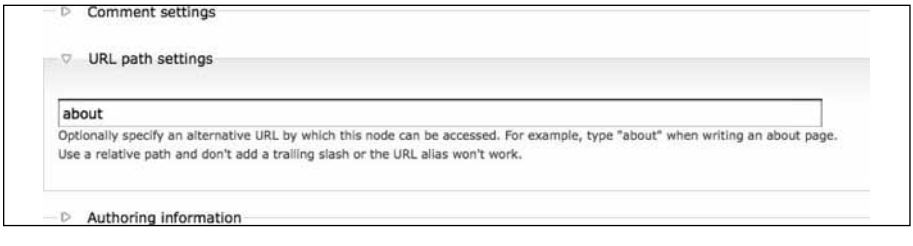


Рис. 2.25. *Ввод синонима пути для узла*

5. Если теперь щелкнуть на ссылке About Us (О компании) в меню еще раз, вы должны увидеть, что адрес URL изменился на *http://www.example.com/about*.
6. Повторите предыдущие действия, чтобы добавить синоним URL для страницы «Home» (Главная), используя home в качестве значения поля URL path settings (Настройки адресов).

Настройка синонимов URL вручную может оказаться утомительным занятием. Модуль Pathauto (домашняя страница проекта: *http://drupal.org/project/pathauto*), о котором будет расска-

зываться в главе 5, позволит определить правила настройки, которые автоматически будут генерировать удобочитаемые адреса URL для всего информационного наполнения веб-сайта (*http://example.com/content/about*), пользователей (*http://example.com/user/admin*) и других элементов.

Модуль Administration Menu

Вероятно, вы уже заметили, что перемещение по страницам администрирования Drupal является немного утомительным занятием. Например, чтобы попасть на страницу администрирования модулей, требуется щелкнуть на ссылке Administer (Администрирование), затем на ссылке Site building (Конструкция сайта), а затем на ссылке Modules (Модули). Существует ли более быстрый способ навигации? К счастью, такой способ существует: это модуль меню администратора Administration Menu (*http://drupal.org/project/admin_menu*), изображенный на рис. 2.26. Этот модуль помещает динамическую полосу раскрывающихся меню в верхней части всех страниц для пользователей, обладающих правами администратора, позволяя легко и просто перемещаться по сайту. Этот модуль не входит в состав ядра системы, поэтому его сначала необходимо загрузить.

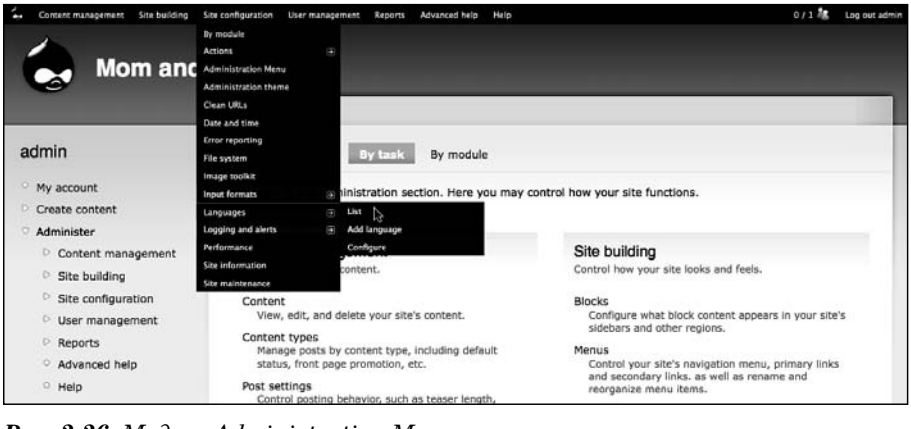


Рис. 2.26. *Модуль Administration Menu*

1. Перейдите на страницу проекта модуля Administration Menu по адресу *http://drupal.org/project/admin_menu*. Найдите таблицу с выпущенными версиями, как показано на рис. 2.27.
2. Найдите версию, отмеченную, как «Recommended for 6.x» (Рекомендуется для версии 6.x) (на рис. 2.27 это будет версия 6.x-1.0), и щелкните на ссылке Download (Загрузить).
3. Загрузив файл, который будет иметь примерное имя *admin_menu-6.x-1.0.tar.gz*, извлеките его содержимое. Файлы модуля должны находиться в папке с именем *admin_menu*.

Releases

Official releases	Date	Size	Links	Status
6.x-1.0	2008-Jun-26	25.43 KB	Download · Release notes · Edit	Recommended for *6.x* ✓
5.x-2.6	2008-Jul-15	28.04 KB	Download · Release notes · Edit	Recommended for *5.x* ✓
4.7.x-1.4	2007-Sep-06	11.17 KB	Download · Release notes · Edit	Recommended for *4.7.x* ✓

Development snapshots	Date	Size	Links	Status
7.x-1.x-dev	2008-Aug-16	26.35 KB	Download · Release notes · Edit	Development snapshot ⊗
6.x-1.x-dev	2008-Aug-08	26.32 KB	Download · Release notes · Edit	Development snapshot ⊗
5.x-2.x-dev	2008-Aug-13	29.22 KB	Download · Release notes · Edit	Development snapshot ⊗

Рис. 2.27. *Выпущенные версии модуля Administration Menu*

4. В своей файловой системе перейдите в папку *sites/all*, где была установлена система Drupal. При необходимости создайте подкаталог *modules*, чтобы текущий путь выглядел как *sites/all/modules*.
5. Переместите папку *admin_menu* в папку *sites/all/modules*. По окончании каталог сайта должен выглядеть, как показано на рис. 2.28.

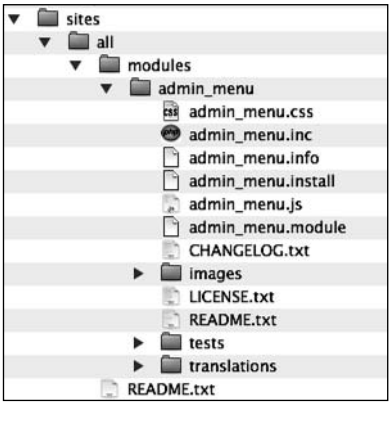


Рис. 2.28. *Файлы модуля Administration Menu в каталоге сайта*

6. Переписав файлы модуля, можно выполнить следующий шаг: установить их на стороне Drupal.

С этого момента во всех практических разделах, где будет возникать необходимость в сторонних модулях, предполагается, что эти шаги уже выполнены и файлы модуля уже находятся в каталоге *sites/all/modules*. Исходные тексты примеров, поставляемые вместе с книгой, уже содержат все модули, необходимые

для создания веб-сайтов в следующих главах. Если вы пожелаете самостоятельно выполнить примеры, приводимые в следующих главах, вам необходимо загрузить и извлечь все необходимые модули подобным способом. Краткий список модулей и тем, используемых в каждой главе, приводится в приложении С.

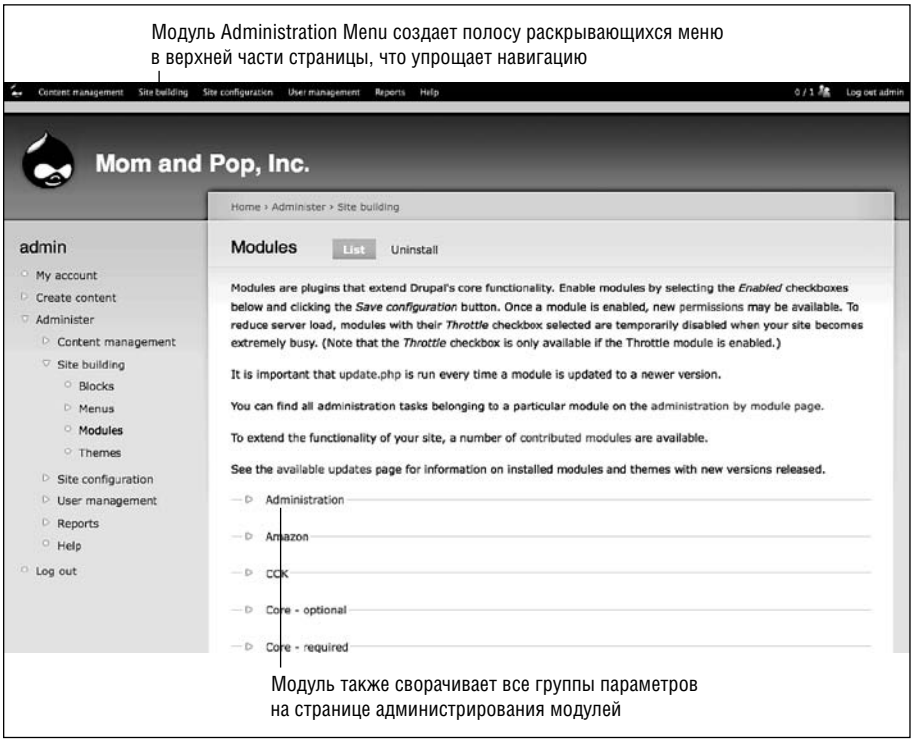


Рис. 2.29. *Действие модуля Administration Menu*

7. В Drupal вернитесь опять на страницу администрирования модулей Administer (Администрирование)→Site building (Конструкция сайта)→Modules (Модули) (*admin/build/modules*).
8. Отыщите группу настроек Administration (Администрирование) и отметьте флажок Enabled (Включено) для модуля Administration Menu.
9. Щелкните на кнопке Save configuration (Сохранить конфигурацию) внизу формы.
10. Вы тут же заметите два признака действия модуля, как показано на рис. 2.29. В верхней части страницы появится горизонтальная полоса меню для выполнения административных задач.
11. Практически любой модуль, устанавливаемый вами, будет обладать своими параметрами настройки, которые можно использовать для

управления поведением модуля и порядком взаимодействия с ним, и модуль Administration Menu не исключение.Перейдите на страницу Administer (Администрирование) (*admin*) и щелкните на вкладке By module (По модулям) (*admin/by-module*), чтобы увидеть список всех активных модулей и параметры настройки, предоставляемые в системе Drupal. Как можно видеть на рис. 2.30, модуль Administration Menu предлагает несколько дополнительных параметров: мы можем определить права на доступ к модулю (что мы сделаем в следующем практическом разделе), мы можем ознакомиться со справочной информацией или перейти на страницу с настройками модуля Administration Menu.

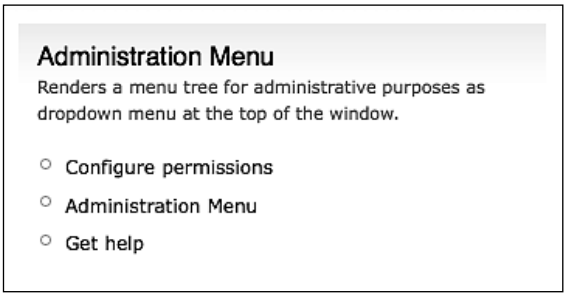


Рис. 2.30. *Параметры настройки модуля Administration Menu*

12. На странице с настройками модуля Administration Menu щелкните на ссылке Get help (Справка) (*admin/help/admin_menu*), чтобы получить справку о модуле. Здесь приводится дополнительная информация о модуле и ссылка на страницу с его настройками.
13. Щелкните на ссылке Administration Menu (*admin/settings/admin_menu*) и посмотрите, какие параметры настройки предлагаются здесь. Помимо всего прочего вы можете установить флажок, отвечающий за сворачивание групп параметров на странице администрирования модуля, что поможет экономить экранное пространство после того, как вы ознакомитесь с пакетами различных модулей

Теперь, когда вы знаете, как устанавливать, включать и настраивать модули, посмотрим, как управлять правами доступа к модулю.

В центре внимания: права доступа

Одна из самых сильных особенностей Drupal заключается в богатстве и избирательности системы управления правами доступа, основанной на таких понятиях, как пользователи, роли и разрешения.

Пользователь

Посетитель веб-сайта. Пользователем может быть кто угодно: случайный посетитель веб-сайта, президент вашей компании, который еже-

дневно ведет свой блог на сайте, ваш системный администратор или кто-то еще, кто вообще не работает в вашей компании, но наделен правом добавления содержимого (например, на сайте в социальных сетях).

Роль

Группа, к которой могут быть отнесены пользователи. Роли могут быть самыми разными, например, «администратор» или «сотрудник коммерческого отдела». По умолчанию в Drupal имеются две роли – «anonymous user» (анонимный пользователь) (присваивается всем пользователям, не зарегистрировавшимся на сайте) и «authenticated user» (зарегистрированный пользователь) (присваивается всем зарегистрировавшимся пользователям), но вы можете создать столько различный ролей, сколько пожелаете.

Разрешения

Нечто, что могут (или не могут) сделать пользователи с определенной ролью на веб-сайте. Каждый модуль может иметь собственный список разрешений. Примерами разрешений могут быть: «иметь доступ к содержимому сайта» и «редактирование собственного блога». Если пользователь не имеет соответствующих разрешений на выполнение некоторых действий, при попытке получить доступ к данной функциональности он будет получать страницу «Access denied» (Отказ в доступе) с сообщением об ошибке.

Страницы «Access denied» (Отказ в доступе) и «Page not found» (Страница не найдена) с сообщениями об ошибках можно настроить на странице Administer (Администрирование)→Site configuration (Настройка сайта)→Error reporting (Сообщения об ошибках) (*admin/settings/error-reporting*).

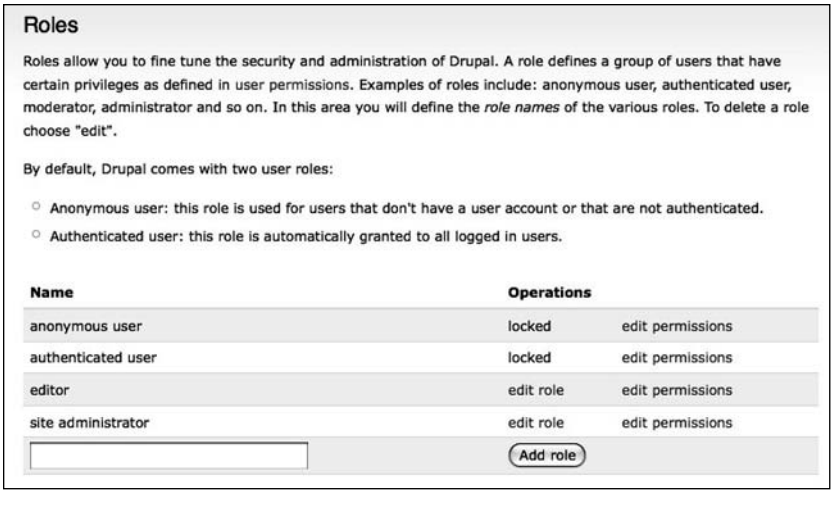
Приступая к разработке любого нового проекта, в первую очередь стоит задуматься, какие группы пользователей будут посещать сайт и какие действия они будут выполнять. Все это будет соответствовать ролям и разрешениям в системе. Попробуйте представить пользователей как можно в более широком смысле. Часто, особенно для маленьких сайтов в небольших организациях, возникает соблазн создать отдельную роль для каждого пользователя (например, «роль Грега»). Но позднее, когда Грега заменит Сьюзан, такой подход будет вызывать сложности, не говоря уже о путанице. Вместо этого подумайте, что будет делать Грег на веб-сайте, например настраивать сайт, обновлять или создавать резервные копии, и назовите роль в соответствии с выполняемыми действиями (например, «администратор сайта»).

Настройка прав доступа пользователей

Управление правами доступа пользователей делится на две части (необязательно): создание одной или более ролей в соответствии с группами посетителей вашего веб-сайта и выдача разрешений для этих ролей.

На странице Administer (Администрирование)→User management (Управление пользователями)→Roles (Роли) (*admin/user/roles*), изображенной на рис. 2.31, можно создавать, изменять и удалять роли. На этой стадии только определяются имена ролей. Отдельным пользователям роли могут присваиваться либо в профилях пользователей, либо на странице администрирования пользователей Administer (Администрирование)→User management (Управление пользователями)→Users (Пользователи) (*admin/user/user*). Для создания и присваивания ролей необходимо обладать разрешением «administer users» (администрировать пользователей).

Щелчок на этой странице на ссылке edit permissions (изменить права) в строке с выбранной ролью вызовет отображение разрешений только для этой роли. Это может быть удобно, когда необходимо определить несколько различных разрешений для отдельной роли.



Roles

Roles allow you to fine tune the security and administration of Drupal. A role defines a group of users that have certain privileges as defined in user permissions. Examples of roles include: anonymous user, authenticated user, moderator, administrator and so on. In this area you will define the *role names* of the various roles. To delete a role choose "edit".

By default, Drupal comes with two user roles:

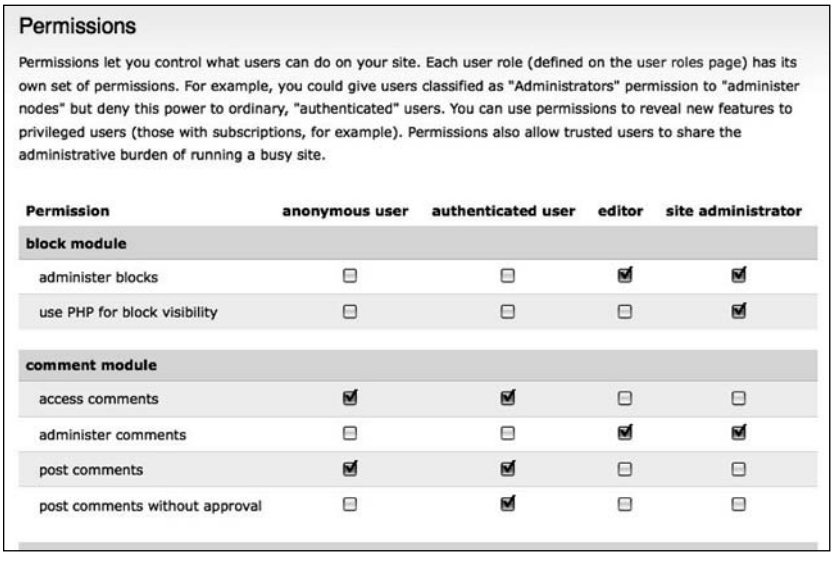
- Anonymous user: this role is used for users that don't have a user account or that are not authenticated.
- Authenticated user: this role is automatically granted to all logged in users.

Name	Operations	
anonymous user	locked	edit permissions
authenticated user	locked	edit permissions
editor	edit role	edit permissions
site administrator	edit role	edit permissions
	Add role	

Рис. 2.31. *Страница администрирования ролей*

При создании новой роли ей не присваиваются никакие разрешения. Вся ответственность за выдачу разрешений новой роли возлагается на администраторов сайта.

Отдельные разрешения ролям можно присвоить на странице Administer (Администрирование)→User management (Управление пользователями)→Permissions (Разрешения) (*admin/user/permissions*), как изображено на рис. 2.32. Доступ к этой странице контролируется разрешением «administer user permissions» (управлять правами доступа), благодаря этому нет необходимости повышать привилегии для пользователей, решающих задачи администрирования пользователей.



Permissions

Permissions let you control what users can do on your site. Each user role (defined on the user roles page) has its own set of permissions. For example, you could give users classified as "Administrators" permission to "administer nodes" but deny this power to ordinary, "authenticated" users. You can use permissions to reveal new features to privileged users (those with subscriptions, for example). Permissions also allow trusted users to share the administrative burden of running a busy site.

Permission	anonymous user	authenticated user	editor	site administrator
block module				
administer blocks	☐	☐	☑	☑
use PHP for block visibility	☐	☐	☐	☑
comment module				
access comments	☑	☑	☐	☐
administer comments	☐	☐	☑	☑
post comments	☑	☑	☐	☐
post comments without approval	☐	☑	☐	☐

Рис. 2.32. *Страница администрирования разрешений*

Важность проверки прав доступа

Обязательно создавайте хотя бы одного «тестового» пользователя для каждой создаваемой вами роли и испытывайте их права доступа к соответствующим разделам сайта. Для учетной записи, создаваемой в ходе установки, известной также как «пользователь № 1», в системе не выполняется проверка прав доступа. Хотя эта особенность удобна при конструировании сайта, при тестировании от имени пользователя 1 будут скрыты ситуации, которые для «простых смертных» посетителей приведут к появлению страницы с сообщением об ошибке «Access denied» (Отказ в доступе).

Для проверки прав доступа нового пользователя выйдите и снова зарегистрируйтесь как пользователь с проверяемой ролью. Вы можете также держать несколько окон броузера открытыми, зарегистрировавшись в каждом из них под разными тестовыми учетными записями. Чтобы получить возможность переключаться между разными учетными записями, не выходя из системы, можно использовать очень удобный блок «Switch user» (Переключить пользователя) модуля Devel (*http://drupal.org/project/devel*).

Также необходимо отметить, что включение и выключение модулей может приводить к изменению разрешений пользователя, поэтому всегда проверяйте страницу с разрешениями после установки или обновления модуля. Кроме того, обязательно проверяйте, что никто не имеет привилегий больше, чем требуется!

Все роли, отличные от «anonymous user» (анонимный пользователь), получают разрешения роли «authenticated user» (зарегистрированный пользователь) плюс разрешения дополнительных присвоенных им ролей. На рис. 2.32 показано, что пользователи с ролью editor (редактор) и site administrators (администраторы сайта) наследуют разрешение access comments (иметь доступ к комментариям), потому что они, само собой разумеется, должны зарегистрироваться (то есть стать зарегистрированными пользователями), чтобы получить возможность работать с сайтом. Однако разрешение «administer comments» (управлять комментариями) является повышенной привилегией, которая не дается зарегистрированным пользователям, поэтому, чтобы обе эти роли получили данное разрешение, его необходимо ролям «editor» (редактор) и «site administrators» (администраторы сайта) присвоить.

Профили пользователей

Для каждого пользователя в системе Drupal имеется страница, которая называется *профилем пользователя*. Это та страница, которая по-

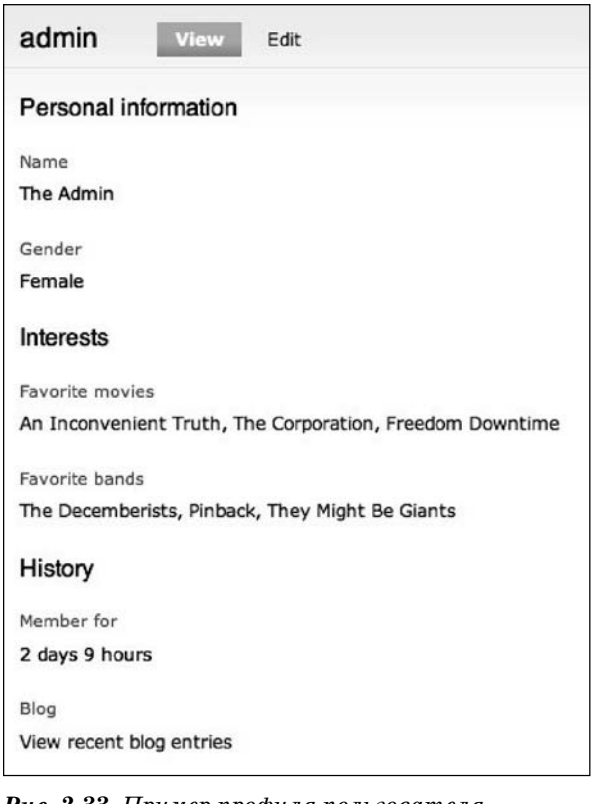


Рис. 2.33. *Пример профиля пользователя*

является после того, как вы зарегистрируетесь на сайте и щелкнете на ссылке My account (Мой профиль). Другие пользователи также могут посетить страницу с вашим профилем, щелкнув на вашем имени пользователя, которое следует за сообщением в блоге или за комментарием, сделанным вами. По умолчанию на странице профиля перечислена информация об учетной записи, такая как имя пользователя и как давно пользователь зарегистрирован на сайте. Однако с помощью дополнительных модулей, включая базовый модуль Profile, можно получать дополнительные сведения, как показано на рис. 2.33, и даже полностью изменять вид страницы. Пользователи могут изменять основные настройки в своих профилях, такие как пароль и часовой пояс, а с помощью других модулей – добавлять дополнительные возможности, такие как выбор языка или поле для выгрузки изображения, отображаемого рядом с каждым сообщением, оставляемым пользователем.

Настройки пользователей

На странице Administer (Администрирование)→User management (Управление пользователями)→User settings (Параметры регистрации) (*admin/user/settings*) имеется множество параметров настройки пользователей, включая:

- Различные параметры регистрации, включая настройку, которая определяет, могут ли посетители сами создавать учетные записи или эта возможность будет предоставлена только администраторам.
- Точный текст различных системных сообщений, отправляемых по электронной почте, например после регистрации пользователя или после того, как его учетная запись будет заблокирована.
- Поддержку подписей, дающую пользователям возможность определять короткий текст, который будет включаться в конец их комментариев.
- Поддержку изображений, дающую пользователям возможность выгружать на сайт изображение, или аватару, которое будет отображаться рядом со всеми их сообщениями и комментариями.

Блокирование пользователей

Сайт с достаточно развитым сообществом и пользующийся определенной популярностью может привлечь внимание неблагожелательных посетителей.

Администраторы с разрешениями «administer users» (администрировать пользователей) могут изменять статус пользователя на Blocked (Блокирован), чтобы лишить их возможности входа на сайт. После этого заблокированные пользователи могут получить права только анонимного пользователя. Кроме того, учетные записи можно полностью удалять, хотя предпочтительнее учетные записи блокировать, а не удалять, чтобы имя пользователя оставалось рядом с его сообщениями.

В Drupal предусмотрена возможность автоматизированного блокирования, настройки которой доступны на странице Administer (Администрирование)→User management (Управление пользователями)→Access rules (Правила доступа) (*admin/user/rules*), которая поможет блокировать (или наоборот, разрешить) вход по имени пользователя, адресу электронной почты или имени хоста. Типичный пример – блокирование имен пользователей, содержащих оскорбительные слова, или предотвращение регистрации посетителей, пользующихся бесплатными почтовыми службами, такими как Hotmail.

Практика: создание ролей

Выше мы говорили о системе управления доступом в Drupal и о том, что она опирается на такие понятия, как пользователи, роли и разрешения, которые определяют, кто будет посещать сайт и какие действия они будут выполнять. Теперь потратим минуту на обсуждение потребностей компании «Mom and Pop, Inc.» в этой области.

Этот сайт будут посещать четыре группы пользователей:

- *Случайные посетители*, для которых будет достаточно иметь возможность отыскивать и читать содержимое, оставлять комментарии после предварительной проверки администратором и отправлять сообщения с контактной информацией по электронной почте. Эти возможности соответствуют правам встроенной роли «anonymous user» (анонимный пользователь).
- *Клиенты*, которые будут регистрироваться на сайте и смогут комментировать его содержимое, но не будут иметь возможность публиковать на нем новости. Поскольку клиенты будут регистрироваться, мы используем для них встроенную роль «authenticated user» (зарегистрированный пользователь).
- Сами *владельцы магазина* Майк и Дженни, которые будут вводить новое содержимое и выполнять незначительные работы по ежедневному администрированию веб-сайта. Им необходимо обладать возможностью создавать и администрировать содержимое, просматривать журналы и статистику и в случае необходимости изменять некоторые настройки веб-сайта. Однако, так как они технически не подготовлены, дополнительные параметры настройки должны быть от них скрыты. Мы назовем эту роль «editor» (редактор), так как основная работа пользователей этой группы состоит в том, чтобы добавлять и изменять содержимое сайта.
- Наконец, Голди – *веб-мастер*, которая фактически занимается конструированием сайта, а также обеспечивает техническую сторону работы Майка и Дженни. Это влечет за собой выполнение таких функций, как установка и обновление модулей и настройка дополнительных возможностей веб-сайта. Даже при том, что сама она может работать с учетной записью пользователя № 1, в какой-то момент ей мо-

жет потребоваться привлечь к работе другого члена своей семьи, поэтому заглянем немного вперед и создадим роль на этот случай. Мы назовем эту роль «site administrator» (администратор сайта).

Эти четыре стандартные роли мы будем использовать во всех последующих главах. На вашем собственном сайте Drupal вы можете определить столько ролей, сколько сочтете нужным.

Теперь можно приступать к настройке прав доступа:

1. Начнем с создания двух дополнительных ролей – одной для Майка и Дженни, и другой для Голди, которая будет заниматься конструированием сайта. Перейдите на страницу Administer (Администрирование)→User management (Управление пользователями)→Roles (Роли) (*admin/user/roles*).
2. Введите `editor` в качестве имени роли и щелкните на кнопке Add role (Добавить роль).
3. Введите `site administrator` в качестве имени роли и щелкните на кнопке Add role (Добавить роль). Теперь страница администрирования ролей должна выглядеть, как было показано ранее на рис. 2.31.
4. После создания ролей желательно сразу же создать тестовых пользователей сайта. Перейдите на страницу Administer (Администрирование)→User management (Управление пользователями)→Users (Пользователи) и щелкните на вкладке Add user (Добавить пользователя) (*admin/user/user/create*).
5. Выполните настройки в соответствии с табл. 2.4 и рис. 2.34, и щелкните на кнопке Create new account (Регистрация).

Таблица 2.4. Значения параметров настройки для первых пользователей сайта

Параметр	**Значение**
Username (Имя пользователя)	Jeanne
Email address (Электронный адрес)	*jeanne@example.com*
Password (Пароль)	По вашему выбору, но используйте что-нибудь достаточно безопасное, например «Mom-0-Rama»
Status (Состояние)	Active (Активный, по умолчанию)
Roles (Роли)	Зарегистрированный пользователь (по умолчанию), editor

6. Повторите те же действия для создания пользователей Mike и Goldie, но для пользователя Goldie добавьте обе роли – «editor» и «site administrator».

7. Создайте также пользователя Random Customer (Случайный покупатель), при этом обратите внимание, что этому пользователю не должно быть присвоено ни одной специальной роли.

Рис. 2.34. *Форма с настройками новой учетной записи*

Практика: настройка разрешений

Теперь, когда созданы роли, можно присвоить им некоторые разрешения, чтобы определить, кто и какие действия сможет выполнять на сайте:

1. Перейдите на страницу Administer (Администрирование)→User management (Управление пользователями)→Permissions (Разрешения) (*admin/user/permissions*).
2. Это очень большая страница, как было показано на рис. 2.32, и здесь имеется множество настроек. Мы будем рассматривать эту страницу по одному модулю за раз.
3. Помните, как мы активировали модуль Administration Menu в предыдущем разделе? В настоящий момент только пользователь №1 может видеть полезное раскрывающееся меню в верхней части страницы. Давайте дадим ту же возможность администраторам сайта и ре-

дакторам, определив настройки, как показано в табл. 2.5. Разрешение access administration menu (доступ к меню администрирования) управляет видимостью этого меню, а разрешение display drupal links (показывать ссылки Drupal) определяет, могут ли ресурсы на сайте Drupal.org быть частью меню. Это очень удобная возможность для Голди, но не для Майка и Дженни, так как они не собираются заниматься решением проблем и вопросами администрирования сайта.

Таблица 2.5. Настройка разрешений для модуля Administration Menu

Разрешение: модуль admin_menu	**anonymous user (анонимный пользователь)**	**authenticated user (зарегистрированный пользователь)**	**editor (редактор)**	**site administrator (администратор сайта)**
access administration menu (доступ к меню администрирования)			отмечен	отмечен
display Drupal links (показывать ссылки Drupal)				отмечен

4. В табл. 2.6 показаны настройки разрешений для модуля Block, который определяет два разрешения: administer blocks (управлять блоками) и use PHP for block visibility (использовать PHP для управления видимостью блоками). Разрешение administer blocks (управлять блоками) дает возможность пользователям с указанной ролью выбирать, какие блоки будут включены и где они будут отображаться на странице, а также создавать новые блоки. Так как это полезная возможность для сайта «Mom and Pop», мы присвоим это разрешение роли «editor» (редактор). Мы также присвоим это разрешение роли «site administrator» (администратор сайта), потому что это одна из административных задач. Однако мы не дадим разрешение use of PHP block visibility (использовать PHP для управления видимостью блоками) ни одной из ролей. Эта дополнительная возможность имеет свои особенности в отношении безопасности и потому только пользователь №1 должен обладать правом доступа к ней.

Всякий раз, когда в названии разрешения вам будет встречаться слово «PHP», тщательно подумайте о том, насколько вы доверяете пользователям с данной ролью, прежде чем дать это разрешение. Злонамеренный пользователь, обладающий доступом к PHP, может нанести любой ущерб сайту – от удаления всего содержимого веб-сайта до рассылки спама всем его пользователям и даже вмешиваться в работу других приложений, за пределами Drupal. Будьте осторожны.

Таблица 2.6. Настройка разрешений для модуля Block

Разрешение: модуль block	**anonymous user (анонимный пользователь)**	**authenticated user (зарегистрированный пользователь)**	**editor (редактор)**	**site administrator (администратор сайта)**
administer blocks (управлять блоками)			отмечен	отмечен
use of PHP block visibility (использовать PHP для управления видимостью блоками)				

5. Разрешения модуля Comment, перечисленные в табл. 2.7, более очевидные. Нам необходимо, чтобы все пользователи имели возможность оставлять комментарии, поэтому мы присвоим это разрешение как анонимным, так и зарегистрированным пользователям. Однако только администраторы сайта и редакторы должны иметь возможность администрировать комментарии (разрешение administer comments). Это разрешение дает им доступ к панели администратора, где можно публиковать комментарии, снимать их с публикации, одобрять их, удалять и так далее. Обе группы пользователей, анонимные и зарегистрированные, должны иметь возможность оставлять комментарии, однако комментарии анонимных пользователей должны сначала передаваться в очередь для модерирования, поэтому разрешение post comments without approval (размещать комментарии без проверки) мы дадим только зарегистрированным пользователям.

Таблица 2.7. Настройка разрешений для модуля Comment

Разрешение: модуль comment	**anonymous user (анонимный пользователь)**	**authenticated user (зарегистрированный пользователь)**	**editor (редактор)**	**site administrator (администратор сайта)**
access comments (иметь доступ к комментариям)	отмечен	отмечен		
administer comments (управлять комментариями)			отмечен	отмечен
post comments (размещать комментарии)	отмечен	отмечен		

post comments without approval (размещать комментарии без проверки)		отмечен		

6. Модуль Filters – это еще одно место, где можно породить проблемы, связанные с безопасностью, поэтому разрешение administer filters (управлять фильтрами) мы дадим только роли «site administrator», как показано в табл. 2.8.

Таблица 2.8. Настройка разрешений для модуля Filters

Разрешение: модуль filters	**anonymous user (анонимный пользователь)**	**authenticated user (зарегистрированный пользователь)**	**editor (редактор)**	**site administrator (администратор сайта)**
administer filters (управлять фильтрами)				отмечен

7. Модуль Menu позволяет настраивать меню навигации по сайту. Для Майка и Дженни это удобная возможность, поэтому мы дадим разрешение administer menu (управлять меню) обеим ролям «editor» и «site administrator», как показано в табл. 2.9.

Таблица 2.9. Настройка разрешений для модуля Menu

Разрешение: модуль menu	**anonymous user (анонимный пользователь)**	**authenticated user (зарегистрированный пользователь)**	**editor (редактор)**	**site administrator (администратор сайта)**
administer menu (управлять меню)			отмечен	отмечен

8. С разрешениями модуля Node, как показано в табл. 2.10, мы поступим немного иначе. Модуль Node определяет множество разрешений для каждого типа содержимого: create (создавать), edit own (изменять свое), edit any (изменять любое), delete own (удалять свое) и delete any (удалять любое). Обычно пользователям даются разрешения на доступ только к их собственному содержимому или дается ограниченный доступ только к ограниченному числу типов содержимого. Модуль Node также имеет «специальное» разрешение administer nodes (управлять материалами). Это разрешение дает пользователям право обходить разрешения на создание, изменение, уда-

ление и просмотр любого содержимого на сайте. Такое разрешение должно даваться только доверенным пользователям. Мы дадим это разрешение роли «editor», присвоенной Майку и Дженни, а также роли «site administrator». Это не только позволит им создавать, изменять и удалять любое содержимое на сайте, но и даст им возможность обращаться к основной странице администрирования содержимого Administer (Администрирование)→Content management (Содержание)→Content (Содержимое) (*admin/content/node*). Однако доступ к содержимому должны иметь все пользователи, поэтому мы дадим разрешение access content (доступ к содержанию сайта) как анонимным, так и зарегистрированным пользователям. Так как типы содержимого достаточно сложны в настройке, мы дадим разрешение administer content types (управлять типами материалов) только роли «site administrator». Кроме того, отметим разрешения revert revisions (переключаться между редакциями) и view revisions (просматривать редакции) для ролей «editor» и «site administrator». Подробнее о редакциях мы поговорим в главе 5.

Таблица 2.10. Настройка разрешений для модуля Node

Разрешение: модуль node	**anonymous user (анонимный пользователь)**	**authenticated user (зарегистрированный пользователь)**	**editor (редактор)**	**site administrator (администратор сайта)**
access content (доступ к содержанию сайта)	отмечен	отмечен		
administer content types (управлять типами материалов)				отмечен
administer nodes (управлять материалами)			отмечен	отмечен
revert revisions (переключаться между редакциями)			отмечен	отмечен
view revisions (просматривать редакции)			отмечен	отмечен

По умолчанию разрешение access content (доступ к содержанию сайта) обеспечивает возможность доступа к любому содержимому на сайте. Однако система Drupal может также расширяться многочисленными сторонними модулями для обеспечения более

избирательного управления доступом к определенным узлам. Например, модуль Node Privacy By Role (*http://drupal.org/project/node_privacy_byrole*) может ограничивать доступ, опираясь на роль пользователя, а модуль Organic Groups (*http://drupal.org/project/og*) может ограничивать доступ для определенных коллекций взаимосвязанного содержимого. Полный список модулей управления доступом можно найти по адресу *http://drupal.org/project/Modules/category/74*.

9. Помните модуль Path из предыдущего раздела? Было бы неплохо, если бы Дженни и Майк также могли добавлять синонимы URL для страниц, создаваемых ими. Мы включим возможность создавать и управлять синонимами URL для ролей «editor» и «site administrator», как показано в табл. 2.11.

Таблица 2.11. Настройка разрешений для модуля Path

Разрешение: модуль path	**anonymous user (анонимный пользователь)**	**authenticated user (зарегистрированный пользователь)**	**editor (редактор)**	**site administrator (администратор сайта)**
administer url aliases (управлять синонимами адресов страниц)			отмечен	отмечен
create url aliases (создавать синонимы адресов страниц)			отмечен	отмечен

10. Модуль System в первую очередь управляет доступом к административному разделу сайта. Как показано в табл. 2.12, мы даем роли «site administrator» доступ ко всему, что имеется на сайте, но ограничиваем доступ к некоторым административным страницам для роли «editor».

Таблица 2.12. Настройка разрешений для модуля System

Разрешение: модуль system	**anonymous user (анонимный пользователь)**	**authenticated user (зарегистрированный пользователь)**	**editor (редактор)**	**site administrator (администратор сайта)**
access administration pages (доступ к страницам администрирования)			отмечен	отмечен

Таблица 2.12 (продолжение)

Разрешение: модуль system	**anonymous user (анонимный пользователь)**	**authenticated user (зарегистрированный пользователь)**	**editor (редактор)**	**site administrator (администратор сайта)**
access site reports (отчеты о доступе к сайту)			отмечен	отмечен
administer actions (управлять действиями)				отмечен
administer files (управлять файлами)				отмечен
administer site configuration (управлять настройками сайта)				отмечен
select different theme (выбирать другую тему оформления)				отмечен

11. Мы еще не рассматривали модуль Taxonomy; он будет обсуждаться в разделе «В центре внимания: модуль Taxonomy», тем не менее, заметим, что он рассматривается как содержимое, поэтому мы дадим доступ к нему для ролей «editor» и «site administrator», как показано в табл. 2.13.

Таблица 2.13. Настройка разрешений для модуля Taxonomy

Разрешение: модуль taxonomy	**anonymous user (анонимный пользователь)**	**authenticated user (зарегистрированный пользователь)**	**editor (редактор)**	**site administrator (администратор сайта)**
administer taxonomy (управлять таксономией)			отмечен	отмечен

12. Наконец, мы подошли к разрешениям модуля User, которые приводятся в табл. 2.14. Нам необходимо обеспечить пользователей доступом к информации в профиле пользователя, а пользователям с ролью «editor» – возможность управлять учетными запися-

ми, при этом возможность изменять имена пользователей и управлять разрешениями мы дадим только пользователям с ролью «site administrators».

Таблица 2.14. Настройка разрешений для модуля User

Разрешение: модуль user	**anonymous user (анонимный пользователь)**	**authenticated user (зарегистрированный пользователь)**	**editor (редактор)**	**site administrator (администратор сайта)**
access user profiles (иметь доступ к профилям пользователей)	отмечен	отмечен		
administer permissions (управлять правами доступа)				отмечен
administer users (администрировать пользователей)			отмечен	отмечен
change own username (изменять свое имя пользователя)				отмечен

13. Проверьте разрешения еще раз и щелкните на кнопке Save permissions (Сохранить права доступа), чтобы сохранить выполненные ранее настройки.
14. Теперь настало время выполнить заключительный шаг: тестирование! Щелкните на ссылке Log out (Выйти) и обратите внимание на отсутствие навигационного меню в боковой колонке. При этом вы можете просматривать содержимое сайта, но если попытаться перейти на страницу, такую как *http://example.com/admin*, вы получите сообщение об ошибке «Access denied» (Отказ в доступе).
15. Зарегистрируйтесь как пользователь «Jeanne». Вы должны получить возможность добавлять содержимое и администрировать сайт.
16. Выйдите и снова зайдите, как пользователь «Random Customer». Вы должны увидеть ссылки My account (Мой профиль) и Log out (Выйти), но не больше.
17. По завершении экспериментов зарегистрируйтесь снова, как пользователь с первой учетной записью, созданной при установке Drupal.

В последующих главах (и в исходных текстах, распространяемых с книгой) мы будем называть пользователей по их ролям: «user» (пользователь), «editor» (редактор) «admin» (администратор), соответственно, и каждой учетной записи будем давать пароль «oreilly».

Практика: форма с контактной информацией

Теперь объединим все знания, полученные к настоящему моменту, и создадим форму с контактной информацией, изображенную на рис. 2.35.

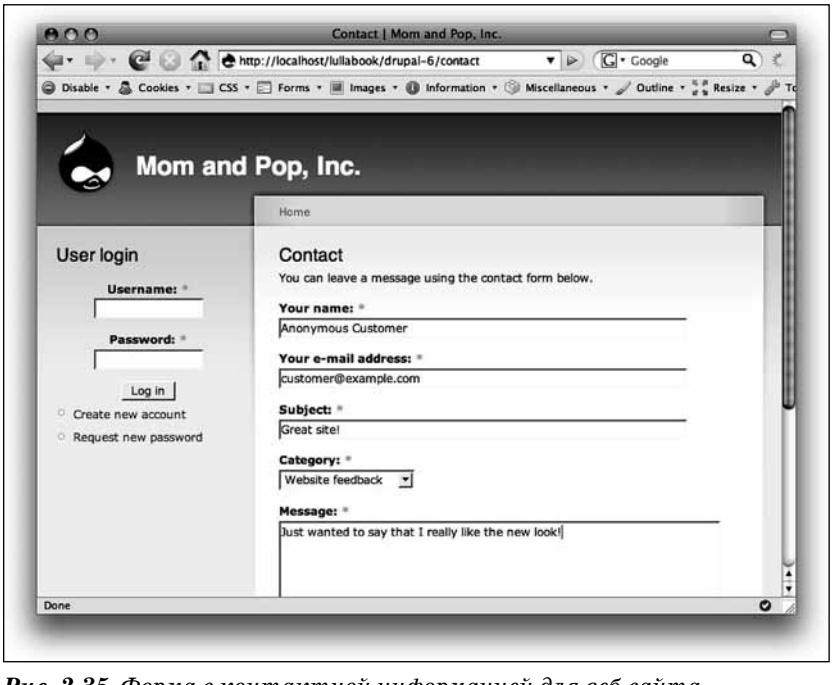


Рис. 2.35. *Форма с контактной информацией для веб-сайта*

1. Сначала включим модуль Contact. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Modules (Модули) (*admin/build/modules*), в разделе Core – optional (Ядро – дополнительные) отметьте флажок в строке с модулем Contact и щелкните на кнопке Save configuration (Сохранить конфигурацию).
2. Далее необходимо определить значения параметров настройки контактной формы. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Contact form (Форма контактов) (*admin/build/contact*).
3. Щелкните на ссылке Add category (Добавить категорию) (*admin/build/contact/add*), введите значения параметров из табл. 2.15, как показано на рис. 2.36, и щелкните на кнопке Save (Сохранить).

Таблица 2.15. Настройки категории контакта

Параметр	Значение
Category (Категория)	Website feedback (Отзыв о веб-сайте)
Recipients (Получатели)	(введите свой электронный адрес)
Auto-reply (Автоматический ответ)	Thanks for sending feedback about our website! We will respond to your inquiry shortly. (Спасибо за отзыв о нашем веб-сайте! Мы ответим на ваш запрос в самое ближайшее время.)

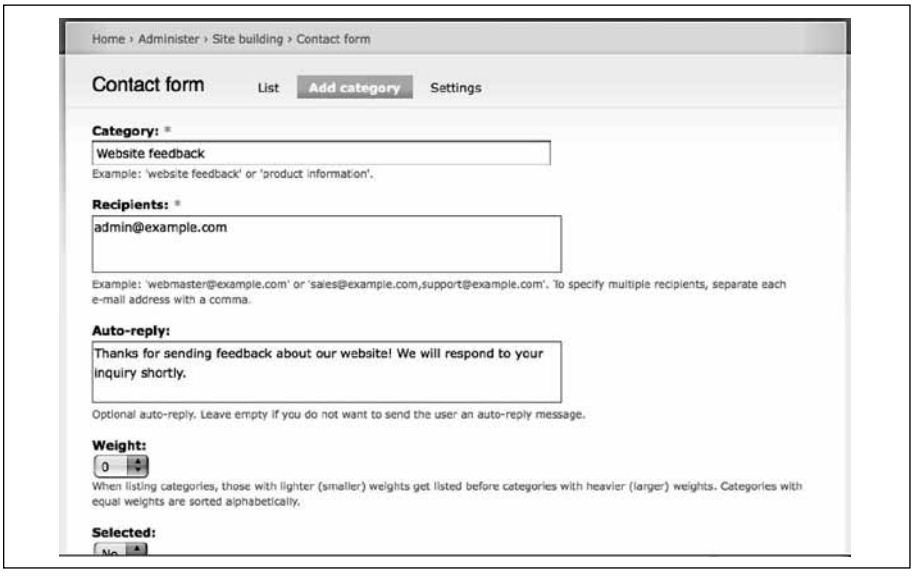


Рис. 2.36. *Настройки формы контактов для категории website feedback (отзыв о веб-сайте)*

4. Повторив те же самые действия, создайте категорию «Sales opportunities» (Возможности сбыта). Для этой категории при желании можно указать другой адрес электронной почты.
5. Далее нам необходимо добавить в навигационное меню сайта ссылку на форму контактов. Этот модуль уже предоставляет пункт меню, готовый к использованию, но по умолчанию он отключен. Если какой-либо модуль предоставляет пункт меню, по умолчанию он будет отображаться в навигационном меню. Однако мы легко можем перенести его в любое другое место. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Menus (Меню)→Navigation (Навигация) (*admin/build/menu-customize/navigation*).

6. Найдите в списке элемент меню Contact (disabled) (Отправить сообщение (выключено)) (он будет окрашен серым цветом). Щелкните на ссылке edit (изменить).
7. Введите настройки из табл. 2.16 и щелкните на кнопке Save (Сохранить).

Таблица 2.16. Настройки пункта меню модуля Contact

Параметр	**Значение**
Menu link title (Название ссылки меню)	Contact (по умолчанию)
Description (Описание)	Get in touch with us (Свяжитесь с нами)
Enabled (Включено)	Отмечено
Expanded (Развернуто)	Не отмечено (по умолчанию)
Parent item (Родительский пункт)	<Primary links> (<Основные ссылки>)
Weight (Вес)	10

8. В заключение необходимо настроить разрешения для контактной формы, чтобы посетители могли использовать ее. Перейдите на страницу Administer (Администрирование)→User management (Управление пользователями)→Permissions (Разрешения) (*admin/user/permission*) и включите разрешения, перечисленные в табл. 2.17. Закончив, щелкните на кнопке Save permissions (Сохранить права доступа).

Таблица 2.17. Разрешения для модуля Contact

Разрешение: модуль Contact	**anonymous user (анонимный пользователь)**	**authenticated user (зарегистрированный пользователь)**	**editor (редактор)**	**site administrator (администратор сайта)**
access site-wide contact form (доступ к контактной форме сайта)	отмечен	отмечен		
administer site-wide contact form (управлять контактной формой сайта)			отмечен	отмечен

9. Наконец, откройте страницу *http://www.example.com/contact*, чтобы увидеть новую контактную форму во всем блеске!

Модуль Contact также предоставляет каждому пользователю сайта его собственную контактную форму, которая доступна на странице просмотра профиля пользователя. Это дает возможность пользователям общаться друг с другом, не сообщая друг другу свои электронные адреса.

В центре внимания: модуль Taxonomy

Мы опробовали практически все основные инструменты, предоставляемые системой Drupal для создания сайтов. Однако чтобы завершить наполнение сайта «Mom and Pop, Inc.» функциональными возможностями, нам необходимо рассмотреть еще одну концепцию: *таксономию*.

Если вы только начинаете изучать Drupal, то наверняка заинтересуетесь, почему повсеместно возникает немного загадочное слово «Таксономия». Не надо пугаться! Это всего лишь технический термин, обозначающий способ организации и классификации. Если вам приходилось сортировать фотографии в семейном альбоме, перемещать сообщения электронной из папки в папку или спорить с другом о принадлежности музыкального коллектива к направлению панк или ска, значит вам уже приходилось иметь дело с таксономией!

Создание таксономии для сайта начинается с того момента, когда вы определите, содержимое каких типов будет присутствовать на сайте и как это можно описать. Например, фотографии можно классифицировать по теме, местоположению, где они были сделаны, и даже по цвету, преобладающему на изображении. В системе Drupal эти категории называются *словарями*. Каждый словарь содержит определенные *рубрики* (такие как Горы, или Автомобили, или Домашние животные), которые могут использоваться для описания содержимого. Всякий раз, когда на сайт отправляется фотография, музыкальный обзор или блог, вы можете выбрать соответствующую рубрику.

Система Drupal поддерживает три типа словарей: простые списки рубрик, древовидные иерархические структуры рубрик и произвольную маркировку («free tagging»), которая позволяют определять новые рубрики по мере создания нового содержимого. Эти словари применяются в разных ситуациях. На рис. 2.37 показано, как каждый тип словаря может использоваться для типа содержимого «product» (продукт).

Для тех, кто не новичок в классификации, отметим, что система таксономии в Drupal также поддерживает дополнительные расширенные особенности, такие как синонимы, связанные рубрики и иерархии с множественным наследованием. Вот так!

После отправки на сайт элемента содержимого названия всех рубрик, куда был отнесен этот элемент, будут отображаться на странице этого элемента в виде ссылок, как показано на рис. 2.38. Каждая из этих ссы-

лок ведет на страницу, где перечислены материалы, отнесенные к этой рубрике, наряду со ссылкой на ленту RSS, на которую посетители могут подписаться, чтобы получать извещения при появлении новых материалов в этой рубрике.

Каждая рубрика имеет уникальный идентификационный номер, который отображается на выделенный адрес URL в системе, такой как *http://example.com/taxonomy/term/3*. Допускается возможность комбинирования рубрик, например, если предположить, что рубрика «политика» имеет идентификационный номер 3, а рубрика «юмор» – 4, то страница с адресом URL *http://example.com/taxonomy/term/3,4* будет отображать материалы, отнесенные сразу к этим двум рубрикам, а страница с адресом URL *http://example.com/taxonomy/term/3+4* будет отображать материалы, отнесенные к одной из двух рубрик. Книга, отмеченная как принадлежащая рубрикам «юмор, роман», не появится в первом списке, но появится во втором.

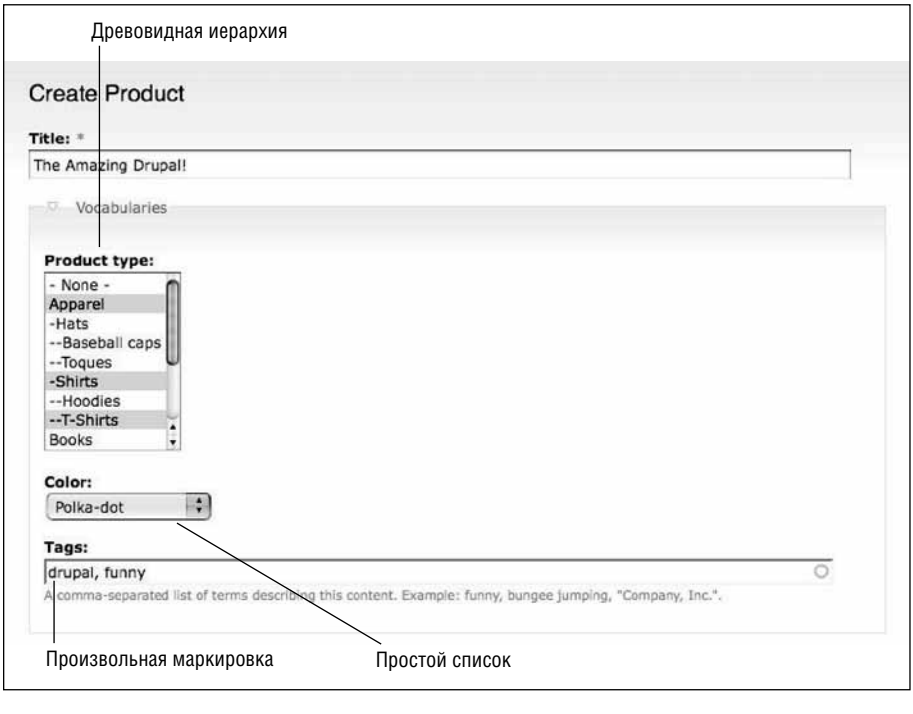


Рис. 2.37. *Примеры типов таксономии*

Система таксономии обладает невероятно широкими возможностями и является одной из самых сильных сторон Drupal как системы управления содержимым. В дополнение к возможностям, присутствующим по умолчанию, имеется несколько весьма интересных сторонних мо-

дулей, таких как Taxonomy Menu (*http://drupal.org/project/taxonomy_menu*), превращающий словарь в меню Drupal, которое может быть помещено в меню среди основных или дополнительных ссылок.

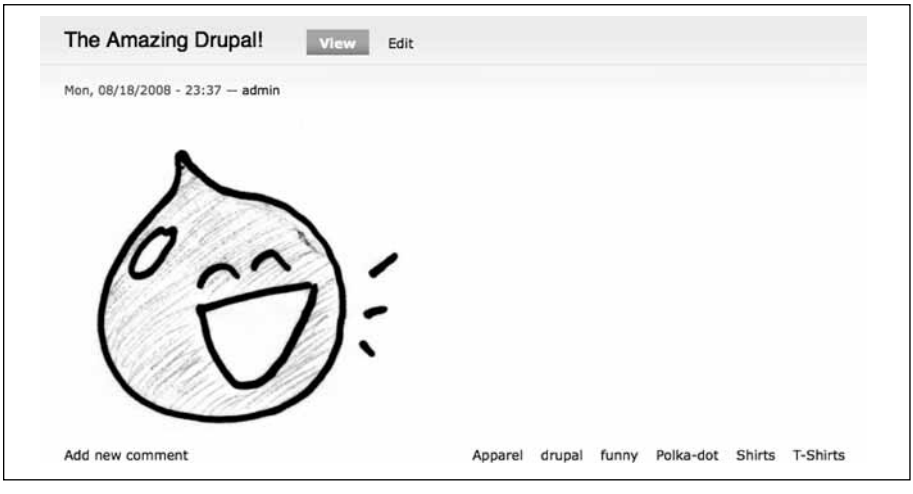


Рис. 2.38. *Пример применения таксономии к материалу*

Практика: блог

Пришло время добавить последнюю функциональную возможность сайта: настроить блоги для Дженни и Майка. Нам необходимо предусмотреть классификацию сообщений в блогах по главным темам: сообщения о событиях, специальные рекламные акции и информация для сообщества. Но блоги должны также обеспечивать возможность «маркировки» с помощью произвольных ключевых слов, которые могут не согласовываться с главными темами.

1. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Modules (Модули) (*admin/build/modules*) и включите базовый модуль Blog. По окончании щелкните на кнопке Save configuration (Сохранить конфигурацию). В результате будет добавлен новый тип содержимого с названием «Blog entry» (Запись в блог) для сообщений, отправляемых Майком и Дженни.
2. Как правило, всякий раз, когда включается новый модуль, необходимо перейти на страницу Administer (Администрирование)→User management (Управление пользователями)→Permissions (Разрешения) (*admin/user/permissions*), чтобы настроить права доступа для модуля. Мы могли бы пойти и посмотреть, какие разрешения добавились к общему списку, но в данном случае нет никакой необходимости устанавливать новые разрешения. И Дженни, и Майк уже обладают разрешением «administer nodes» (управлять материалами), которое

автоматически дает им права, предоставляемые модулем Blog, поэтому мы можем к этому не возвращаться.

3. Прежде чем создавать какие-либо блоги, давайте сначала настроим таксономию сайта. Перейдите на страницу Administer (Администрирование)→Content management (Содержание)→Taxonomy (Таксономия) (*admin/content/taxonomy*).
4. Сначала необходимо создать словарь для классификации новых элементов, которые будут отправляться. Щелкните на вкладке Add vocabulary (Добавить словарь) (*admin/content/taxonomy/add/vocabulary*), введите значения параметров из табл. 2.18, как показано на рис. 2.39, и щелкните на кнопке Save (Сохранить).

Блог или не блог?

Когда начинающие конструкторы сайтов хотят создать персональный блог на сайте, они часто используют модуль Blog, входящий в состав Drupal. Однако иногда это не самый лучший выбор. Чтобы понять, отвечает ли этот модуль вашим потребностям, посмотрим, какие особенности он несет:

- Новый тип содержимого, который называется «Blog» (Блог)
- Обзорную страницу со всеми блогами на сайте по адресу *http://www.example.com/blog*.
- Обзорную страницу для каждого пользователя со всеми сообщениями в блоге по адресу *http://www.example.com/blog/1*, где 1 – это идентификационный номер учетной записи.
- Ссылки внизу каждого сообщения, которые позволяют получить список всех сообщений данного автора.
- Лента RSS для каждого пользователя в блоке, которая рассылает сообщения в блоге данного пользователя.
- Частная ссылка My Blog (Мой блог) в навигационном меню для каждого пользователя, имеющего право создавать сообщения в блоге.

Если вы конструируете сайт, основное содержимое которого составляют сообщения одного пользователя, все дополнительные ссылки, предоставляемые модулем Blog, будут только раздражать. Намного проще и удобнее создать собственный тип содержимого с названием «сообщение в блоге» и оставить модуль Blog отключенным. Если вы конструируете сайт, где эти сообщения будут являться лишь одной из многих разновидностей содержимого и сообщения будут отправляться многими пользователями, тогда модуль Blog будет отличным выбором. По этой причине мы будем использовать его на сайте «Mom and Pop, Inc.»!

Таблица 2.18. Настройки словаря «Category»

Параметр	Значение
Identification (Идентификация)	
Vocabulary name (Название словаря)	Category
Description (Описание)	пустое (по умолчанию)
Help text (Справочный текст)	пустое (по умолчанию)
Content types (Типы материалов)	
Content types (Типы материалов)	Blog entry (Запись в блоге)
Settings (Настройки)	
Tags (Теги)	не отмечено (по умолчанию)
Multiple select (Множественный выбор)	не отмечено (по умолчанию)
Required (Обязательно)	отмечено

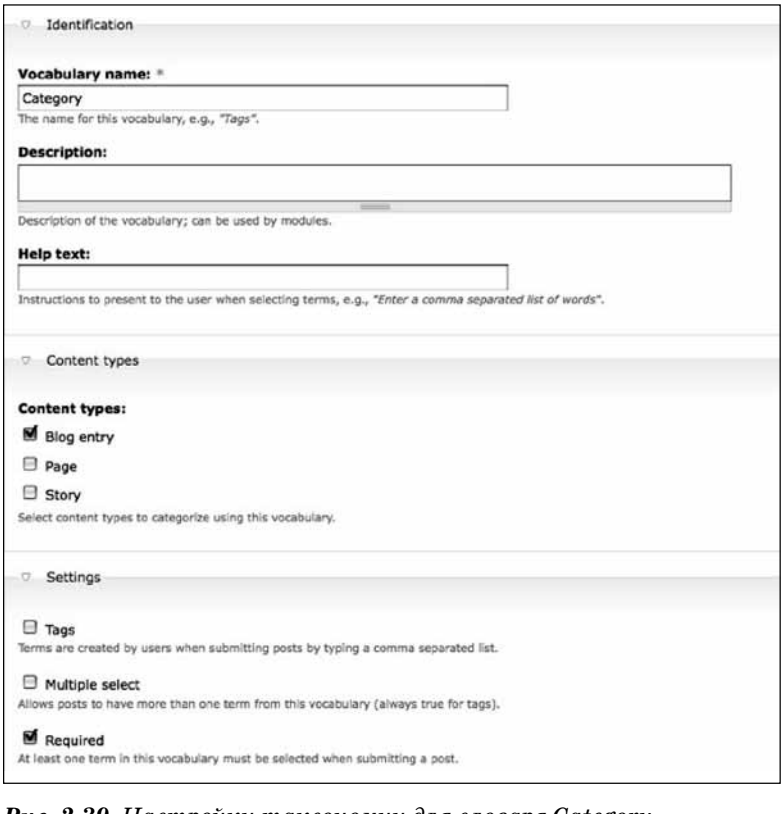


Рис. 2.39. *Настройки таксономии для словаря Category*

5. Сохранив словарь, вы опять вернетесь на главную страницу администрирования таксономии. Щелкните на ссылке add terms (добавить термины) рядом с названием словаря Category и добавьте несколько названий рубрик, таких как «In-Store Event» (События в магазине), «Special Deals» (Специальные акции), «Community» (Сообщество) и «Website Information» (Информация на сайте).
6. Затем добавьте словарь с названием «Tags» (Теги) для дополнительной информации, представляющей, например, определенные продукты. Так же, как и прежде, вернитесь на страницу Administer (Администрирование)→Content management (Содержание)→Taxonomy (Таксономия) (*admin/content/taxonomy*) и щелкните на вкладке Add vocabulary (Добавить словарь) (*admin/content/taxonomy/add/vocabulary*). Введите значения параметров из табл. 2.19 и щелкните на кнопке Save (Сохранить).

Таблица 2.19. Настройки словаря «Tags»

Параметр	Значение
Identification (Идентификация)	
Vocabulary name (Название словаря)	Tags
Description (Описание)	пустое (по умолчанию)
Help text (Справочный текст)	пустое (по умолчанию)
Content types (Типы материалов)	
Content types (Типы материалов)	Blog entry (Запись в блог)
Settings (Настройки)	
Tags (Теги)	отмечено
Multiple select (Множественный выбор)	не отмечено (по умолчанию)
Required (Обязательно)	не отмечено (по умолчанию)

7. Откройте страницу Create content (Создать материал)→Blog entry (Запись в блог) (*node/add/blog*), чтобы увидеть форму с настройками блога, с добавленной поддержкой таксономии, как показано на рис. 2.40. Теперь создайте запись в блоге.
8. В окончательном виде персональный блог пользователя будет выглядеть, как показано на рис. 2.41. Здесь можно видеть ссылки, с помощью которых можно оставлять свои комментарии, и просматривать список всех сообщений, принадлежащих только пользователю admin, или все сообщения на сайте, отнесенные к данной рубрике.

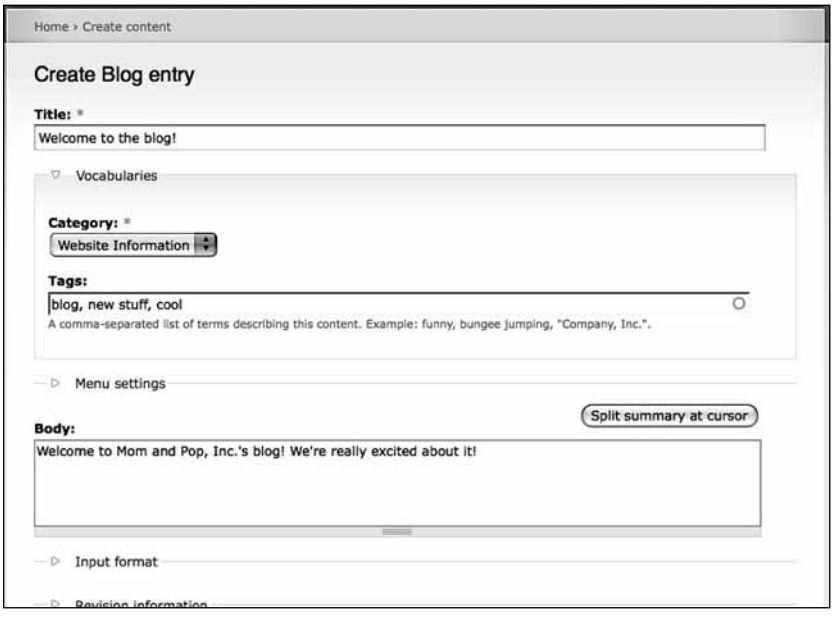


Рис. 2.40. *Форма настройки блога с включенной поддержкой таксономии*

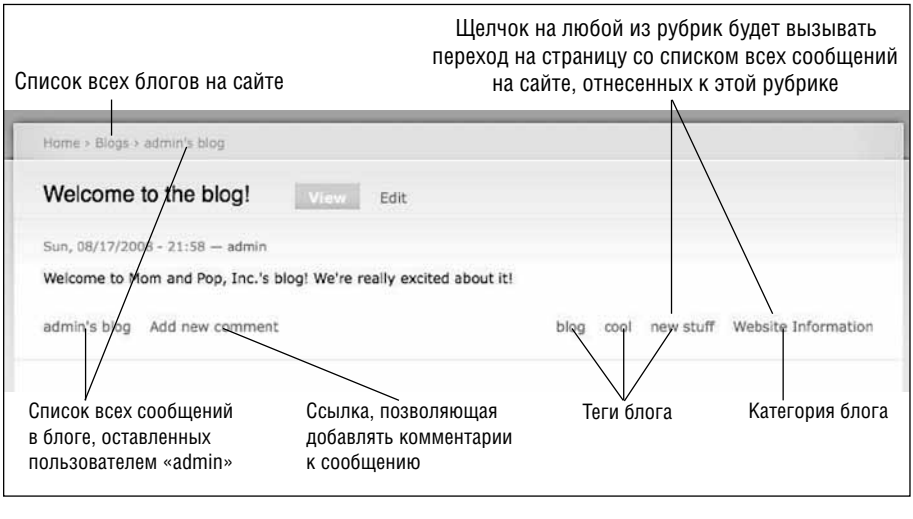


Рис. 2.41. *Окончательный вид блога*

9. Теперь, закончив настройку блога, добавим ссылку на блог в основное меню. Как и модуль Contact, модуль Blog по умолчанию предоставляет пункт меню. Перейдите на страницу Administer (Администриро-

вание)→Site building (Конструкция сайта)→Menus (Меню)→Navigation (Навигация) (*admin/build/menu-customize/navigation*). Отыщите элемент Blogs (disabled) (Блоги (отключено)) и щелкните на ссылке edit (изменить).

10. Введите значения параметров из табл. 2.20 и щелкните на кнопке Save (Сохранить).

Таблица 2.20. Настройки пункта меню модуля Blog

Параметр	Значение
Menu link title (Название ссылки меню)	Blog
Description (Описание)	View our blog (Смотрите наш блог)
Enabled (Включено)	отмечено
Expanded (Развернуто)	не отмечено (по умолчанию)
Parent item (Родительский пункт)	<Primary links> (<Основные ссылки>)
Weight (Вес)	0 (по умолчанию)

11. Чтобы сделать отправку комментариев более удобной для посетителей сайта, следует внести пару незначительных изменений. Чтобы изменить настройки комментариев, перейдите на страницу Administer (Администрирование)→Content management (Содержание)→Content types (Типы материалов) (*admin/content/types*) и щелкните на ссылке edit (изменить) для типа Blog entry (Запись в блог) (*admin/content/node-type/blog/edit*).

12. Раскройте группу параметров Comment settings (Установки комментариев). Ух ты! Сколько параметров. Большинство из этих параметров не представляют для нас интереса и имеют отношение к отображению комментариев, но пара параметров будет нам полезна. Например, мы можем дать анонимным пользователям возможность оставлять вместе с комментарием свое имя и адрес своей домашней страницы, мы можем сделать необязательным предварительный просмотр комментариев (как и в случае с узлами) и мы можем обеспечить отображение формы ввода комментария непосредственно рядом с сообщением в блоге, а не на отдельной странице. Введите значения параметров из табл. 2.21 и щелкните на кнопке Save content type (Сохранить тип материала).

13. Вернитесь к списку блогов, щелкнув на ссылке Blog (Мой блог) в меню навигации по сайту, которая была создана нами ранее.

Таблица 2.21. Настройки комментариев в блоге

Параметр	Значение
Anonymous commenting (Анонимные комментарии)	Anonymous posters may leave their contact information (Анонимные пользователи могут оставлять контактную информацию)
Preview comment (Просмотр комментариев перед отправкой)	Optional (Необязательно)
Location of comment submission form (Расположение формы отправки комментариев)	Display below post or comments (Показывать ниже сообщения или комментарии)

14. Щелкните на ссылке Log out (Выйти), чтобы стать анонимным пользователем, затем щелкните на ссылке Blog (Мой блог) в меню и затем щелкните на заголовке сообщения в блоге, чтобы перейти к просмотру полного текста сообщения Welcome (Добро пожаловать). На странице немедленно должна появиться форма ввода комментария вместе с полями для ввода имени, электронного адреса и адреса домашней страницы, как показано на рис. 2.42. После отправки комментария вы получите сообщение, что комментарий поставлен в очередь для утверждения.

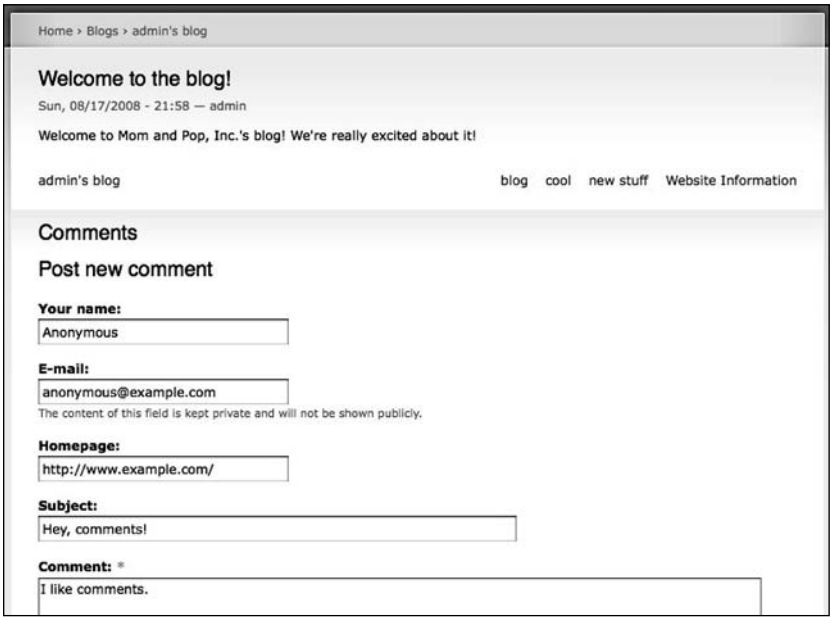


Рис. 2.42. *Форма ввода комментария для анонимных пользователей после настройки*

15. По окончании эксперимента зарегистрируйтесь снова, как пользователь admin. Теперь вы можете опубликовать комментарий, перейдя на страницу Administer (Администрирование)→Content management (Содержание)→Comments (Комментарии), на вкладку Approval queue (Очередь на подтверждение) (*admin/content/comment/approval*).

В центре внимания: инструменты модерирования содержимого

После того как вы откроете перед своими пользователями ворота к возможности стать активными участниками процесса добавления содержимого, перед вами неизбежно встанет проблема модерирования содержимого, то есть придется предпринимать меры, чтобы оскорбительное, вульгарное содержимое, непредусмотренная реклама и спам неизменно оставались за пределами сайта.

Предотвратить появление такого рода содержимого можно с помощью двухступенчатого подхода: задействовать механизм автоматизированного определения спама и удалять спам вручную.

Инструменты автоматизированного определения спама

Существует множество инструментов, специализированных для автоматического определения спама, но два из них в особенной степени достойны упоминания в контексте создания сайтов на базе Drupal: Akismet и Mollom. Оба эти инструмента являются веб-службами и требуют наличия «ключа API», для обеспечения возможности взаимодействий между веб-сайтом и центральным сервером.

Служба Akismet (*http://akismet.com*) была создана Мэттом Мулленвегом (Matt Mullenweg), создателем и руководителем проекта платформы WordPress. Она проверяет содержимое комментариев на вашем сайте и «обратных ссылок» (trackbacks) (выдержки из сообщений в других блогах, ссылающихся на вас) и на основе результатов анализа содержимого миллионов других блогов либо удалит содержимое, если оно определяется как спам, либо передаст его дальше, если это не спам. Установив модуль Akismet, вы сможете использовать эту службу, опирающуюся на коллективный опыт миллионов других блогов. Велика вероятность, что к тому моменту, когда спамер обучит свой сценарий автоматизированной рассылки отправлять спам на ваш сайт, служба Akismet уже будет иметь «отпечатки пальцев» нападающего и сможет удалять спам еще до того, как кто-либо увидит его на вашем сайте. Для редких исключений модуль Akismet предоставляет возможность вручную помечать содержимое как спам, и впоследствии использует эту информацию. С момента основания в 2005 году служба Akismet отфильтровала более семи миллиардов комментариев со спамом и зарекомендовала себя как проверенное решение. Модуль Akismet можно загрузить по

адресу *http://drupal.org/project/akismet*, а ключ API для доступа к службе можно получить по адресу *http://wordpress.com/api-keys/*.

Более молодая служба Mollom (*http://mollom.com*) была запущена Бенджамином Шраувеном (Benjamin Schrauwen) и Дризом Байтаертом (Dries Buytaert), создателем и руководителем проекта Drupal. Она очень похожа на службу Akismet, выполняя автоматическую проверку содержимого комментариев и предотвращая саму возможность передачи спама на веб-сайт. Как и служба Akismet, она опирается на коллективный опыт других веб-сайтов, где установлен модуль Mollom, а кроме того, служба Mollom совместима со многими другими системами управления содержимым и языками программирования.

Однако служба Mollom предпринимает попытки преодолеть некоторые недостатки службы Akismet следующими способами:

- Поддерживает возможность блокировки спама не только в комментариях, но и в форме с контактной информацией, в формах узлов (блоги, темы форумов и так далее), в формах регистрации пользователей и в других. Это уникальная особенность Mollom, отсутствующая в конкурирующих решениях, что делает эту службу «универсальным решением на все случаи», позволяя избежать использования одной службы для проверки комментариев, а другой – для обработки форм регистрации.
- Отличает спам от доброкачественного содержимого, а в «пограничных случаях» отображает тест CAPTCHA (Completely Automated Public Turing test to tell Computers and Humans Apart – полностью автоматизированный публичный тест Тьюринга для различения компьютеров и людей), искаженное изображение символов, которые пользователь должен ввести вручную. Этот тест позволяет человеку продолжить работу, но блокирует действия спам-роботов. Для обеспечения максимальной доступности тест CAPTCHA отображается как изображение и воспроизводится как аудиофайл. В отличие от Akismet, Mollom ликвидирует потребность в очередях на модерирование и тем самым уменьшает бремя модерирования, лежащее на плечах администраторов сайта, – спам блокируется еще до того, как он попадет на сайт.
- Позволяет удалять не только спам, но и низкопробное или оскорбительное содержимое, а также содержимое, не связанное с обсуждаемой темой. Кроме того, модуль Mollom возвращает оценку качества для каждого сообщения, опираясь на правила правописания, язык и правила пунктуации, что позволит поддерживать минимальный уровень профессионализма на вашем сайте.
- Использует мощь единой системы OpenID, возвращая «показатель репутации» для учетных записей в системе OpenID по всем веб-сайтам. Это гарантирует немедленную передачу сообщений, принадлежащих людям, и блокировку спамерских сообщений, если они рассылаются с использованием учетной записи OpenID.

- Генерирует график, показывающий объем спама, как показано на рис. 2.43.

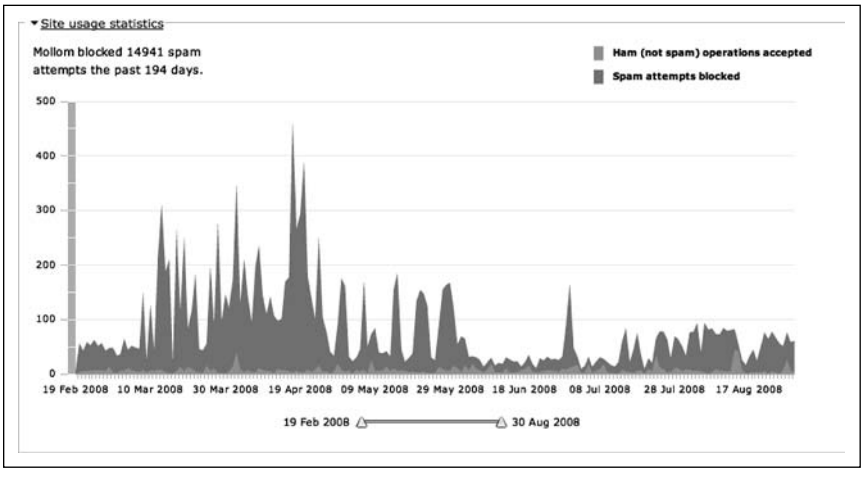


Рис. 2.43. *Пример отчета по спаму, генерируемого модулем Mollom*

Цель создания службы Mollom состоит в том, чтобы полностью ликвидировать необходимость проверки содержимого вручную, используя в «неясных» случаях тест CAPTCHA. В отличие от многих других сайтов, пользователи будут вынуждены проходить тест CAPTCHA, только если их сообщения находятся «на границе» допустимого и недопустимого, но не для каждого сообщения, как это реализовано в других решениях, таких как модуль CAPTCHA (*http://drupal.org/project/captcha*).

Модуль Mollom доступен для загрузки по адресу *http://drupal.org/project/mollom*, а ключ API можно получить, создав учетную запись на сайте *http://mollom.com*.

Инструменты модерирования содержимого вручную

Автоматизированные инструменты прекрасно блокируют очевидный спам и действия роботов, но как воспрепятствовать на нашем сайте действиям троллей, рассылающим порнографию или другое оскорбительное содержимое? Модуль Mollom предоставляет возможность помечать такое содержимое, но на некоторых сайтах предпочтение отдается более «трудоемкому» решению, особенно если могут наступить правовые последствия при появлении оскорбительного содержимого на веб-сайте хотя бы на секунду.

Ядро системы Drupal включает некоторые основные инструменты модерирования содержимого, такие как возможность устанавливать признак «не опубликовано» по умолчанию для содержимого любого типа (и тем самым полностью скрывать это содержимое от всех, кроме автора

и администраторов) и управление редакциями, что позволяет при последующих правках выполнять «откат» до одной из одобренных ранее версий. Но на многих сайтах, построенных на базе Drupal, подходят к решению проблемы с двух сторон, используя модуль modr8 (*http://drupal.org/project/modr8*), который добавляет признак «moderation queue» (ожидает проверки) к содержимому и некоторые весьма неплохие возможности предварительного просмотра, а также модуль Revision Moderation (*http://drupal.org/project/revision_moderation*) – простой вспомогательный модуль, который гарантирует, что при внесении изменений опубликованной останется предыдущая, одобренная редакция содержимого.

В центре внимания: темы оформления

Темы определяют внешний вид сайта на базе Drupal. Это не то, что гарантирует функциональность сайта, однако темы могут придать ему *внешний вид*, отличающий его от других сайтов.

Поиск тем оформления

На сайте Drupal.org имеется огромный репозиторий свободно распространяемых тем оформления, созданных сторонними разработчиками. Перечень этих тем можно найти на странице *http://drupal.org/project/Themes*, а опробовать их «вживую» – на странице *http://themegarden.org*.

Качество тем оформления в репозитории Drupal варьируется в довольно широких пределах. Эти темы создавались для самых разных целей и потребностей, сторонними разработчиками с различным уровнем подготовки в области программирования и дизайна. Загрузите несколько тем и обязательно прочитайте их файлы *README.txt*, чтобы узнать, как их задействовать. Многие из этих тем могут использоваться как заготовки, на основе которых администраторы могут создать собственное представление своих сайтов.

Кроме того, в экосистеме Drupal появилось несколько компаний, предлагающих темы оформления на коммерческой основе. Как правило, эти темы имеют более высокое качество, обладают более высокой надежностью и могут предоставлять для администраторов более простую настройку.

Установка тем оформления

Установка новых тем оформления выполняется практически так же, как установка модулей. Нужно загрузить файл архива *.tar.gz* с темой со страницы проекта и извлечь ее. Поместить каталог с темой в каталог *sites/all/themes*, после чего новая тема должна появиться в системе Drupal на странице Themes administration (Темы оформления), как показано на рис. 2.44.

Как и в случае с модулями, темы оформления, созданные для Drupal 5, не совместимы с версией Drupal 6, и наоборот. Темы, созданные для Drupal 5, не появятся на этой странице, если вы используете Drupal 6.

Настройка тем оформления

Переключать темы можно на странице управления темами Administer (Администрирование)→Site building (Конструкция сайта)→Themes (Темы оформления) (*admin/build/themes*), изображенной на рис. 2.44.

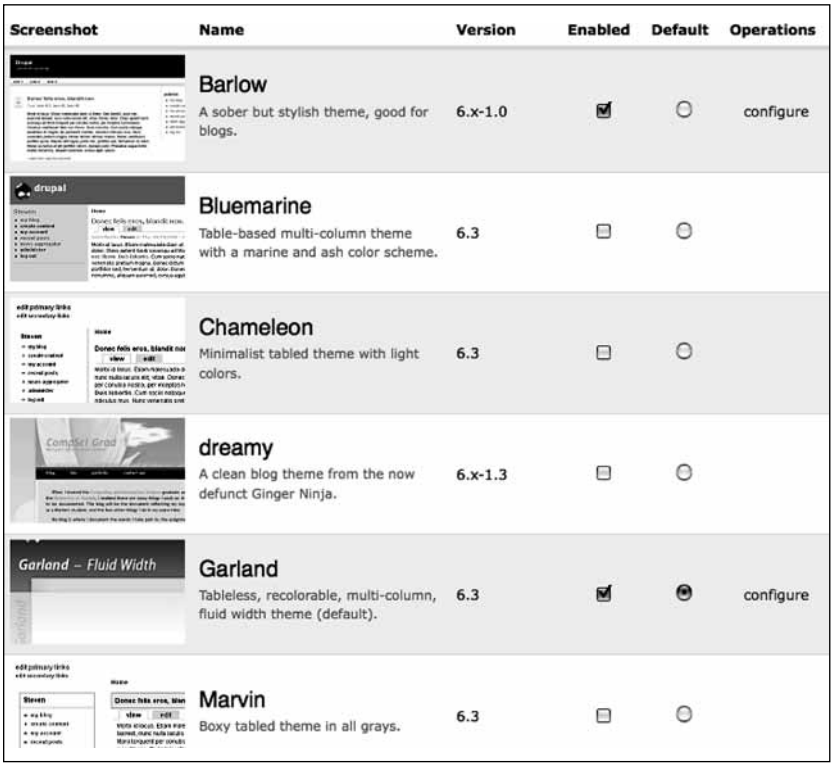


Рис. 2.44. *Страница управления темами*

В большинстве случаев администраторы могут игнорировать колонку Enabled (Включено) – отметка в колонке Default (По умолчанию) фактически определяет тему, которую будут видеть все пользователи, не имеющие права выбирать тему оформления. Чтобы включить новую тему, достаточно просто установить соответствующий флажок в колонке Default (По умолчанию) и щелкнуть на кнопке Save configuration (Сохранить конфигурацию) внизу формы. Новая тема оформления сайта будет задействована немедленно.

Колонка Enabled (Включено) позволяет администраторам назначить несколько тем, которые будут доступны для выбора пользователям, обладающих разрешением select different theme (выбирать другую тему оформления). На рис. 2.45 показана страница с настройками учетной записи такого пользователя в случае, когда для выбора доступно несколько тем оформления. Без этого разрешения, которое по умолчанию не выдается, пользователи не увидят эти варианты выбора. Немногие сайты дают своим пользователям это разрешение, потому что, в конце концов, ваша тема соответствует вашему фирменному стилю и является вашей дизайнерской разработкой.

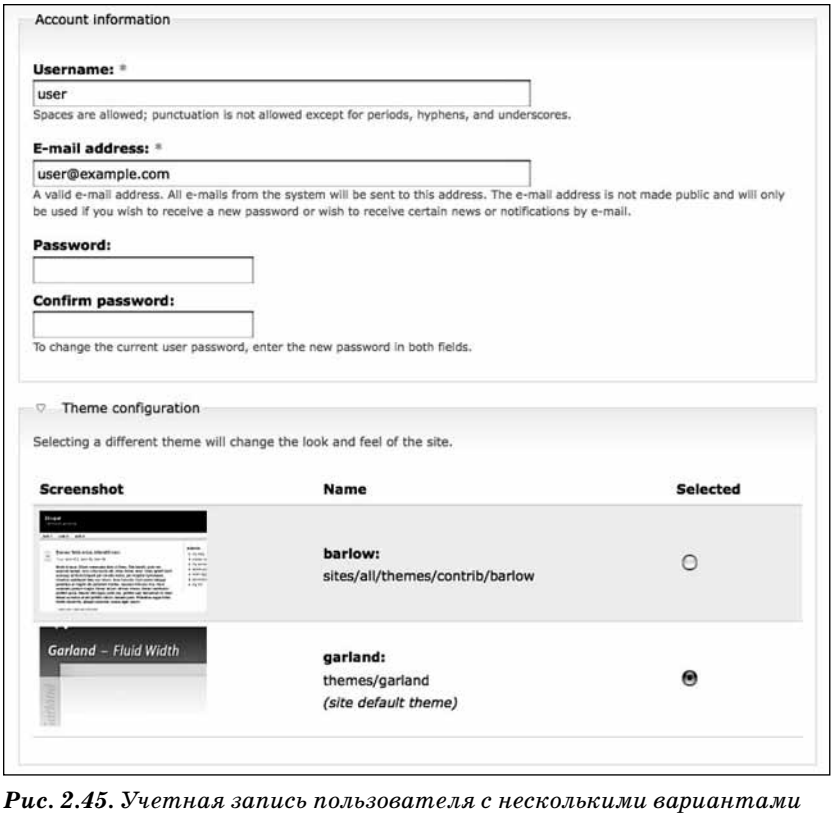


Рис. 2.45. *Учетная запись пользователя с несколькими вариантами оформления, доступными для выбора*

Система Drupal предлагает массу параметров настройки, которые могут использоваться темами. Существует два способа настройки тем оформления. Чтобы определить «глобальные» значения параметров, применяемых ко всем темам, выберите вкладку Global settings (Общие настройки) на странице Administer (Администрирование)→Site building (Конструкция сайта)→Themes (Темы оформления)→Configure (Настроить) (*admin/build/themes/settings*). Чтобы определить настройки для определенной темы

оформления или установить значения параметров, предлагаемых только определенной темой, щелкните на ссылке configure (настроить) в колонке Operations (Действия) рядом с названием темы. Настройки, которые вы увидите, могут отличаться от темы к теме.

Разрешение select different theme (выбирать другую тему оформления) может играть роль полезного инструмента отладки. Администратор сайта может включить обычную тему оформления сайта и базовую тему, такую как Garland, и переключаться между ними, когда на сайте обнаруживаются ошибки. Если ошибка обнаруживается только при использовании обычной темы, следовательно, в первую очередь ошибку следует искать в реализации темы. Если ошибка проявляется в обеих темах оформления, следовательно, ошибка зарыта где-то глубже, например в базе данных или в настройках.

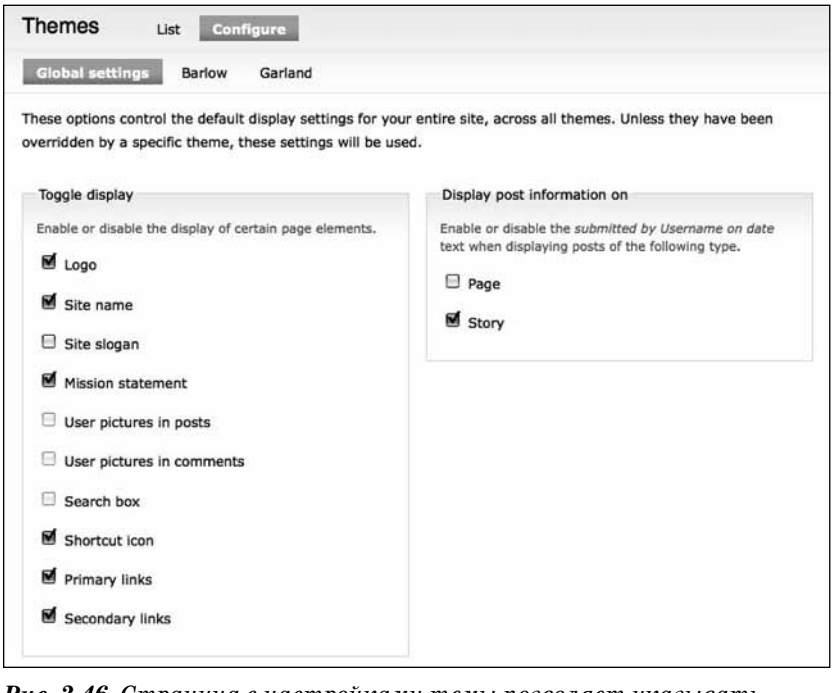


Рис. 2.46. *Страница с настройками темы позволяет указывать, какие элементы страницы будут отображаться*

На страницах с настройками имеется возможность управлять отображением элементов темы, включая изображение логотипа, название сайта, его лозунг, заявление о целях и другие, как показано на рис. 2.46. Некоторые из флажков могут быть заблокированы некоторыми другими настройками системы. В этом случае Drupal не дает никаких подсказок, но в нашем примере параметры User pictures in posts (Аватары в сообщениях) и User pictures in comments (Аватары в комментариях) заблоки-

рованы потому, что на странице Administer (Администрирование)→User management (Управление пользователями)→User settings (Параметры регистрации) (*admin/user/settings*) выключен параметр Picture support (Поддержка изображений). Параметр Search box (Строка поиска) заблокировано, потому, что не был включен модуль Search.

Кроме того, страница с настройками темы позволяет администраторам выгружать на сайты собственные изображения логотипа и ярлыка (известного также, как ярлык сайта или страницы, который отображается в адресной строке броузера) или просто указывать ссылки на один из их серверов.

Некоторые темы оформления, такие как базовая тема Garland, также используют модуль Color, что позволяет администратору сайта настраивать цветовую схему, используя удобный элемент выбора цвета, реализованный на языке JavaScript. На рис. 2.47 показан модуль Color в действии.

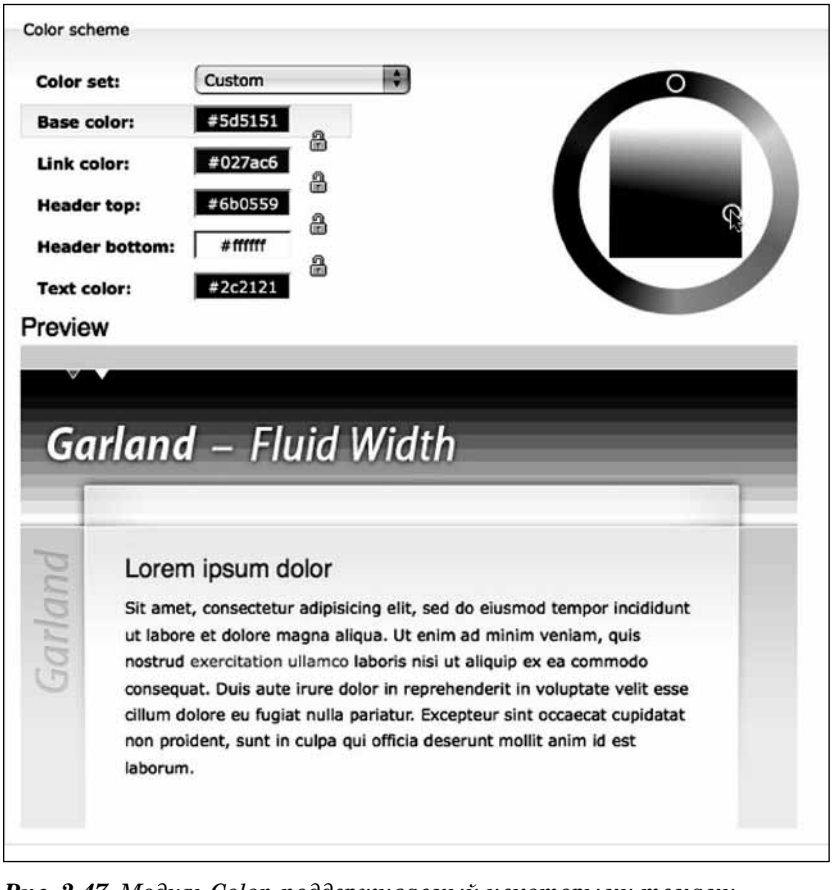


Рис. 2.47. *Модуль Color, поддерживаемый некоторыми темами, дает возможность изменять цветовую схему сайта*

Настройки, такие как site name (имя), site slogan (слоган), mission statement (миссия), определяются на странице Administer (Администрирование)→Site configuration (Настройка сайта)→Site information (О сайте) (*admin/settings/site-information*).

Блоки и области

Важно помнить, что области размещения блоков определяются темой оформления и различные темы могут предлагать разные области. Если у вас имеются блоки, привязанные к определенной области в одной теме, и вы переключаетесь на использование другой темы, в которой отсутствует область с тем же названием, эти блоки станут невидимыми. После переключения на другую тему обязательно посетите страницу управления блоками Administer (Администрирование)→Site building (Конструкция сайта)→Blocks (Блоки) (*admin/build/block*) и посмотрите, какие области имеются в этой теме. Возможно, вам придется привязать блоки к другой области, чтобы получить все преимущества новой темы.

Настройка темы оформления административного раздела

В отличие от многих других систем управления содержимым и программного обеспечения управления блогами, Drupal по умолчанию не предоставляет отдельное оформление административного раздела сайта. В разделе, где выполняется настройка сайта и производится редактирование содержимого, используется то же самое оформление, что и на остальной части сайта. Однако большие формы и таблицы, необходимые для настройки сайта Drupal, часто плохо сочетаются с дизайном и схемой размещения элементов в остальных разделах сайта, что в ре-

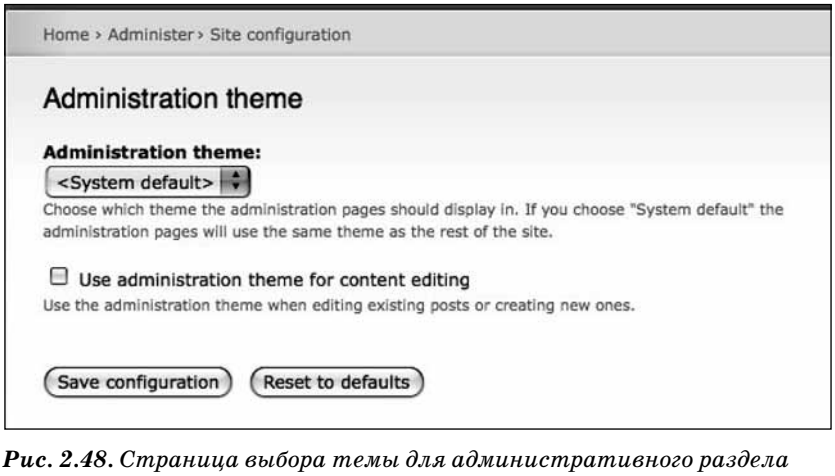


Рис. 2.48. *Страница выбора темы для административного раздела*

зультате приводит к «нарушению» административных страниц. Более того, многие администраторы предпочитают более явно показать пользователю, что он зашел в административный раздел сайта, недоступный для обычных посетителей. По этим причинам Drupal позволяет выбирать тему для административного раздела на странице Administer (Администрирование)→Site configuration (Настройка сайта)→Administration theme (Тема админа) (*admin/settings/admin*). Эта тема будет использоваться для оформления всех административных страниц (путь к которым в строке URL начинается с «admin»), и, при желании, для страниц, где создается и редактируется содержимое сайта. Страница выбора темы для административного раздела показана на рис. 2.48.

Практика: придание сайту неповторимого внешнего вида

Теперь пришло время сделать сайт менее похожим на сайт Drupal и более похожим на сайт компании «Mom and Pop, Inc.». В этом разделе мы займемся настройкой темы, чтобы придать сайту неповторимый внешний вид.

1. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Themes (Темы оформления) (*admin/build/themes*) и убедитесь, что отмечена радиокнопка Default (По умолчанию) для темы Garland. Щелкните на кнопке Save configuration (Сохранить конфигурацию). Теперь вы должны увидеть свой сайт, оформленный темой Garland, если ранее эта тема еще не была задействована.

Если оформление сайта не изменилось в соответствии с вновь выбранной темой, проверьте следующее:

- Возможно, для оформления административного раздела используется отдельная тема. В этом случае *все* административные страницы будут оформляться в соответствии с темой, выбранной для административного раздела. Чтобы увидеть новое оформление, вам необходимо перейти на какую-нибудь другую страницу, например на главную.
- Возможно, на сайте разрешается выбирать оформление из нескольких тем, и ваша учетная запись имеет разрешение select different theme (выбирать другую тему оформления). В этом случае нужно либо снять флажки Enabled (Включено) для неиспользуемых тем на странице Themes (Темы оформления), либо зайти на вкладку Edit (Изменить) на странице My account (Мой профиль) и выбрать новую тему.

2. Щелкните на вкладке Configure (Настроить) (*admin/build/themes/settings*), чтобы изменить глобальные параметры настройки, применяемые ко всем темам.

3. С помощью параметра Logo image settings (Настройки логотипа) выгрузите файл изображения *mom_and_pop_logo.png*, находящийся в каталоге *assets/ch02-basic* среди исходных кодов примеров, распространяемых вместе с книгой. После сохранения формы щелчком на кнопке Save configuration (Сохранить конфигурацию) вы должны увидеть новый логотип в верхнем левом углу. Но на голубом фоне он выглядит ужасно!
4. Щелкните на вкладке Garland (*admin/build/themes/settings/garland*), чтобы перейти к настройкам темы Garland, где имеется встроенный элемент выбора цвета модуля Color.
5. Выберите цветовую схему, наиболее приятную для вашего глаза и сочетающуюся с логотипом. Подойдите к этому творчески! Добившись удовлетворительного результата, щелкните на кнопке Save configuration (Сохранить конфигурацию).
6. Теперь с новым логотипом и названием сайта «Mom & Pop, Inc.» вверху страница выглядит немного беспорядочно. Давайте отключим отображение названия сайта. Снимите флажок Site name (Название сайта) в группе параметров Toggle display (Показывать) и щелкните на кнопке Save configuration (Сохранить конфигурацию) еще раз.

По окончании у вас должен получиться сайт, в котором присутствует привлекательный логотип компании «Mom and Pop, Inc.» в стиле Web 2.0 с собственной, отличительной цветовой схемой, как показано на рис. 2.49.

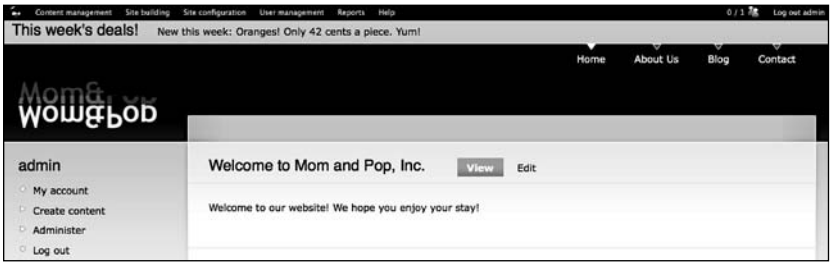


Рис. 2.49. *Веб-сайт, получивший новый логотип и цветовую схему*

В центре внимания: редактирование содержимого и работа с изображениями

Теперь у нас имеется сайт, обладающей всеми функциональными возможностями, которые были затребованы Майком и Дженни, с настроенными правами доступа и привлекательным внешним видом, дополненным отличительной цветовой схемой и с новым логотипом. Однако осталось еще кое-что: необходимо упростить процесс редактирования содержимого и выгрузку изображений на сайт.

Редактирование содержимого

Как уже упоминалось выше в этой главе, по умолчанию в системе Drupal предусмотрен ввод содержимого в формате HTML. Как и большинство землян, Майк и Дженни не обладают знаниями для работы с кодом разметки, поэтому очень важно дать им возможность форматировать содержимое и добавлять изображение без этих знаний. Неудивительно, что за многие годы сообществом было выработано множество решений этой проблемы:

Панели инструментов

Некоторые пользователи прекрасно справляются с разметкой HTML, если им дать панель инструментов, с помощью которой они смогут вставлять нужные им теги. Модуль BUEditor (*http://drupal.org/project/bueditor*), изображенный на рис. 2.50, является примером реализации такой панели инструментов.

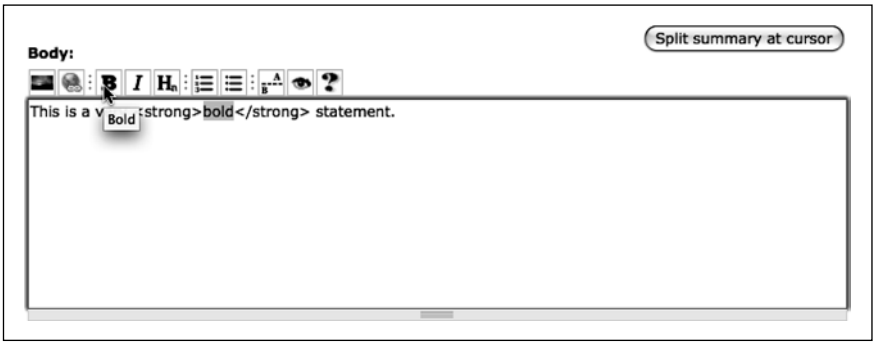


Рис. 2.50. *Модуль BUEditor реализует панель инструментов, помогающую создавать разметку HTML*

Инструменты преобразования текста в разметку HTML

Такие модули, как Textile (*http://drupal.org/project/textile*) и Markdown Filter (*http://drupal.org/project/markdown*), описываемые в главе 5, позволяют преобразовывать простой текст, например `**bold**`, в эквивалентную разметку HTML (`<strong>bold</strong>`). Запомнить и использовать такой синтаксис намного проще, чем синтаксис языка разметки HTML.

Визуальные редакторы

Визуальные редакторы не только предоставляют панель инструментов, но еще и отображают форматирование непосредственно в области ввода текста, как показано на рис. 2.51, напоминая текстовые процессоры. Существует несколько модулей для системы Drupal, реализующих интегрированные визуальные редакторы, но наиболее

популярными являются TinyMCE (*http://drupal.org/project/tinymce*) и FCKeditor (*http://drupal.org/project/fckeditor*).

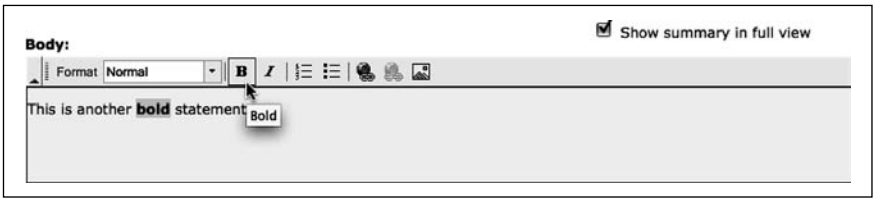


Рис. 2.51. *Модуль FCKeditor, отображающий форму редактирования*

Чем дальше мы уходим от обычной разметки HTML, тем выше вероятность, что «умный» визуальный редактор запутается в сложном форматировании. Различия, имеющиеся между веб-броузерами, несовместимость с каскадными таблицами стилей, используемыми для настройки оформления веб-сайта, и некоторые другие проблемы не такое уж редкое явление. Во многих случаях приходится идти на компромисс, потому что пользователи сайта не заинтересованы в изучении языка HTML только для того, чтобы вывести какой-то текст жирным шрифтом, а какой-то курсивом. Но из-за возможных ловушек часто бывает разумно спросить себя: «А нужно ли это на моем сайте», прежде чем задействовать «замечательный» модуль редактора HTML.

Для Майка и Дженни мы будем использовать визуальный редактор. Несмотря на то, что модули TinyMCE и FCKeditor по своим функциональным возможностям почти эквивалентны, тем не менее, модуль FCKeditor выглядит немного предпочтительнее, в том смысле, что он используется в большинстве сайтов на базе Drupal 6, а его разработчики проявляют более высокую активность в его развитии. Поэтому мы выбрали FCKeditor, хотя имеется выбор из множества других модулей. В списке модулей из категории Filters/editors (Фильтры/редакторы) на странице *http://drupal.org/project/Modules/category/63* перечислены все модули фильтров, включая и визуальные редакторы.

Обратите внимание на модуль WYSIWYG API (*http://drupal.org/project/wysiwyg*). Назначение этого модуля состоит в том, чтобы предоставить один модуль, который будет поддерживать произвольное число визуальных редакторов; он является наиболее вероятным местом появления новшеств в этой области.

Работа с изображениями

Еще одним важным недостатком системы Drupal помимо отсутствия встроенного визуального редактора является отсутствие встроенных инструментов для работы с изображениями. Изначально система Dru-

pal имеет встроенный модуль Upload, позволяющий любому пользователю, обладающему разрешением «upload files» (загружать файлы), прикреплять файлы изображений к создаваемому им содержимому. После этого пользователи могут вручную добавить теги `<img>`, внедряющие эти файлы изображений в содержимое. Однако для многих пользователей это слишком трудоемкий процесс, особенно, когда необходимо вставить множество изображений в большую статью.

Хорошая новость (и, фактически, плохая) состоит в том, что имеется множество сторонних модулей, предоставляющих разнообразные возможности, всю широту которых можно увидеть, ознакомившись со списком модулей в категории Media на странице *http://drupal.org/project/Modules/category/67*. На протяжении прошлых лет было предложено довольно много решений этой проблемы, причем некоторые из них были реализованы достаточно давно и завоевали определенную популярность.

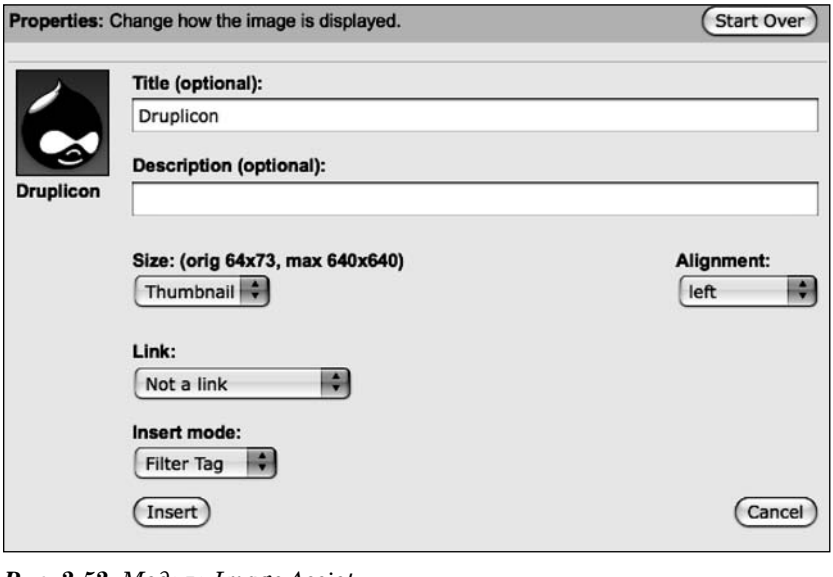


Рис. 2.52. *Модуль Image Assist*

Модуль Image Assist (*http://drupal.org/project/img_assist*), изображенный на рис. 2.52, использует по максимуму возможность Drupal управлять различными типами содержимого: он добавляет каждое изображение, присоединенное к элементу содержимого, как отдельный узел «image». Это упрощает возможность использования других особенностей Drupal, таких как отправка комментариев для каждого изображения или сборка альбомов изображений, использованных в других сообщениях. А до появления модулей ImageAPI (*http://drupal.org/project/imageapi*) и ImageCache (*http://drupal.org/project/imagecache*) (описывае-

мых в главе 7) он был единственным простым способом автоматического создания уменьшенных копий больших изображений.

Более современное решение, которое мы будем рассматривать в этой главе, представляет модуль IMCE (*http://drupal.org/project/imce*), изображенный на рис. 2.53. Он способен работать вместе с нашим визуальным редактором и дает возможность выгружать изображения с помощью всплывающего диалога непосредственно в процессе создания содержимого. Выгрузив изображение, пользователь может вставить его в свое сообщение, используя привычные инструменты визуального редактора. Кроме того, модуль IMCE следит за всеми изображениями, выгруженными пользователем в прошлом. Его можно настроить так, что он будет сохранять изображения, выгруженные каждым пользователем, в отдельных каталогах или помещать их в один общий каталог. Возможность выделения отдельных каталогов для пользователей позволяет им накапливать свои библиотеки изображений и хранить их отдельно от файлов других пользователей. Кроме того, модуль IMCE позволяет накладывать такие ограничения, как размер файла и разрешение изображения, а также устанавливать квоты на объем дискового пространства, которое пользователь может занять на сервере.

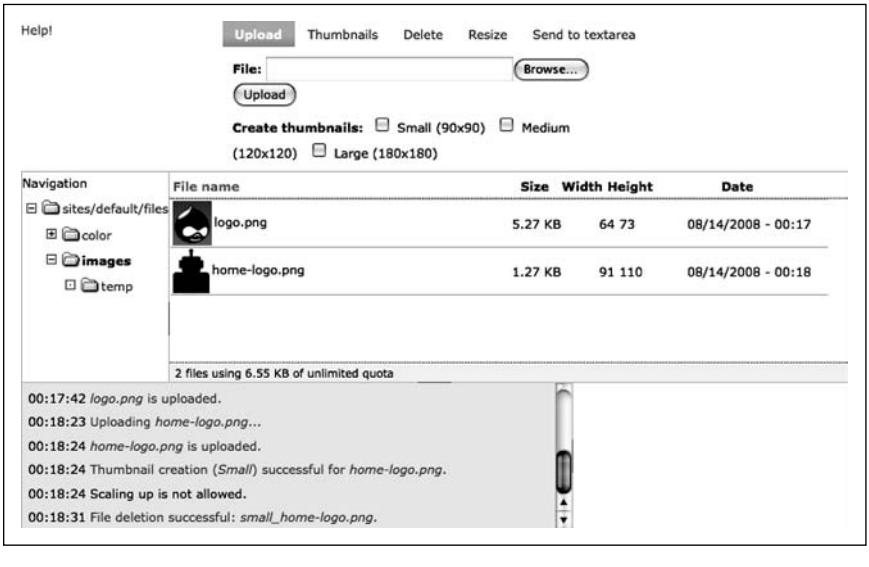


Рис. 2.53. *Модуль IMCE*

В центре внимания: форматы ввода и фильтры

Возможно, вы заметили эту интересную группу параметров настройки материала и тела блока, которую мы упорно игнорировали, – Input formats (Форматы ввода), изображенную на рис. 2.54. *Формат ввода*, вы-

бранный для содержимого, определяет, как это содержимое будет отображаться на сайте. Форматы ввода – это очень важный элемент обеспечения безопасности Drupal, поэтому стоит остановиться, чтобы разобраться с ними. Формат ввода определяет, как будет «интерпретироваться» содержимое и какие изменения в разметку HTML следует внести, прежде чем оно будет передано броузеру для отображения. Формат ввода определяется для каждого элемента содержимого, поэтому система Drupal всегда знает, что искать и какие изменения вносить.

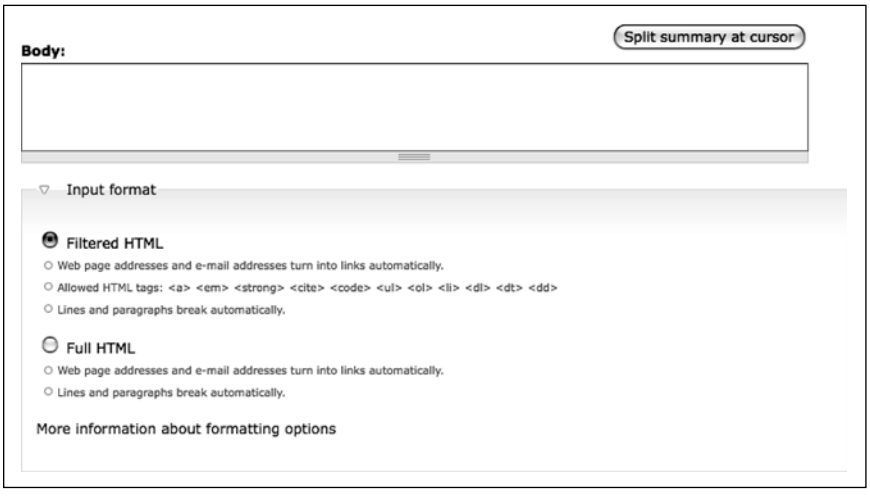


Рис. 2.54. *Форматы ввода для тела узла*

Обратите внимание: какой бы формат по умолчанию ни был выбран на странице администрирования, он будет доступен *всем* пользователям, включая анонимных посетителей. Поэтому всегда лучше использовать формат Filtered HTML (Фильтрованный HTML).

На рис. 2.54 представлено всего два варианта, Filtered HTML (Фильтрованный HTML) и Full HTML (HTML без ограничений). Эти два формата ввода поставляются в составе ядра Drupal по умолчанию. На сайтах может иметься несколько дополнительных форматов ввода, часть которых может предоставляться модулями, такими как фильтр программного кода PHP, а другая часть – создаваться вами.

Доступ к форматам ввода может ограничиваться ролями, поэтому вы можете позволить всем без исключения использовать один формат ввода, подобный Filtered HTML (Фильтрованный HTML), и добавить свой формат ввода, подобный Full HTML (HTML без ограничений), только для пользователей, пользующихся вашим доверием, например, обладающих ролью администратора.

Если пользователь сообщит вам о таинственной пропаже вкладки редактирования узла, проверьте его формат ввода. Система Drupal не разрешит редактировать содержимое, если роль, назначенная пользователю, не дает ему право доступа к формату ввода содержимого. Эту особенность можно использовать для предотвращения возможности редактирования определенных страниц пользователями, которые в противном случае имели бы такую возможность.

Форматы ввода конструируются из *фильтров*. На рис. 2.55 показан список фильтров, использованных для определения формата ввода Filtered HTML (Фильтрованный HTML). Основную работу выполняют фильтры, тогда как формат ввода – это всего лишь группа фильтров. Фильтр изменяет содержимое и дает на выходе разметку HTML, пригодную для отображения. В качестве примеров фильтров можно назвать Line break converter (Преобразователь переводов строки), который преобразует символы возврата каретки в теги `<br />` и `<p>`, и URL filter (Фильтр URL), который преобразует текст с адресом URL, такой как *http://www.example.com*, в гиперссылку `<a href="http://www.example.com">http://www.example.com</a>`. Фильтры могут использоваться для выполнения полезных и интересных преобразований содержимого (например, модуль Pirate [*http://drupal.org/project/pirate*] преобразует текст содержимого, имитируя пиратскую манеру говорить), но по-настоящему важными они становятся, когда речь заходит о безопасности сайта. Люди способны нанести сайту урон, получив в свое распоряжение текстовое поле ввода. Использование фильтров, специально предназначенных для обезвреживания злонамеренного содержимого, таких как фильтр разметки HTML, поможет обезопасить ваш сайт от взлома.

Фильтр разметки HTML является самым важным фильтром – он обезвреживает опасные теги HTML и защищает сайт от различных нападений, заключающихся в том, чтобы обманом внедрить злонамеренный JavaScript или другой выполняемый код. Этот фильтр включен по умолчанию только в формате ввода Filtered HTML (Фильтрованный HTML). Вы должны полностью доверять тем, кто имеет доступ к формату ввода без этого фильтра, такому как Full HTML (HTML без ограничений).

В пределах форматов фильтры упорядочены по присвоенным им «весам» и применение фильтров производится именно в такой последовательности. Порядок следования фильтров по умолчанию для формата ввода Filtered HTML (Фильтрованный HTML) можно увидеть на рис. 2.56. Многие сторонние модули позволяют добавлять дополнительные фильтры, которые можно смешивать в любом порядке, добавляя их в существующие форматы ввода или составляя из них свои собственные форматы. Эта возможность подробнее будет рассматриваться в главе 5, где будет обсуждаться добавление фильтров.

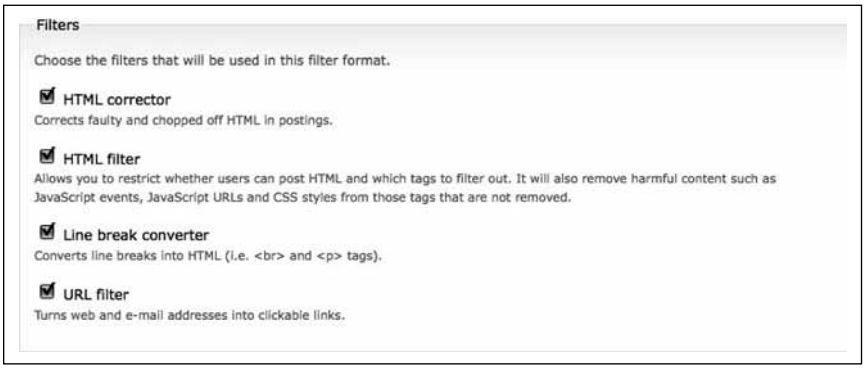


Рис. 2.55. *Фильтры для формата ввода Filtered HTML*

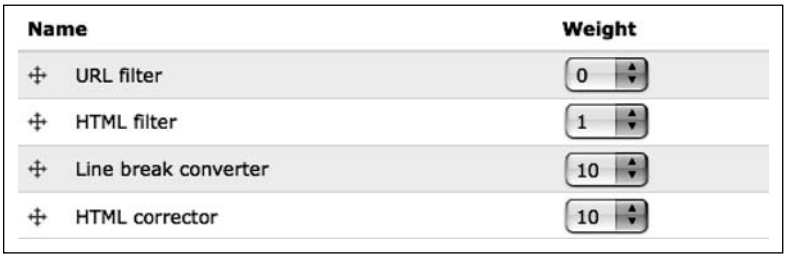


Рис. 2.56. *Переупорядочение фильтров в формате ввода Filtered HTML*

Очень важно понимать, что форматы применяются, только когда содержимое извлекается из базы данных и подготавливается к отображению на странице. Когда пользователь вводит содержимое в форму и сохраняет ее, это содержимое сохраняется в базе данных точно в том виде, в каком его ввел пользователь. Когда посетитель открывает страницу для просмотра, Drupal извлекает информацию из базы данных, применяет соответствующий ей формат, поочередно пропуская через каждый фильтр, и затем отображает окончательный результат в броузере. Схематическое представление этого процесса приводится на рис. 2.57. Обратите внимание, что для получения окончательного результата фильтры применяют свои правила по очереди. Если бы в данном конкретном случае вместо Filtered HTML (Фильтрованный HTML) мы задействовали формат ввода Full HTML (HTML без ограничений), конечный результат был бы немного иным. Благодаря применению формата Filtered HTML (Фильтрованный HTML) на экран выводится текст «alert('I cAn pwn U');», потому что фильтр удалил теги `<script>` перед отображением. Если бы использовался формат Full HTML (HTML без ограничений), эти теги остались бы на месте и вместо отображения текста броузер *выполнил* бы сценарий JavaScript. В данном случае сценарий просто вывел бы довольно безопасное окно JavaScript с надписью «I cAn pwn U», но вообще это мог бы быть злонамеренный сценарий, способный причинить ущерб.

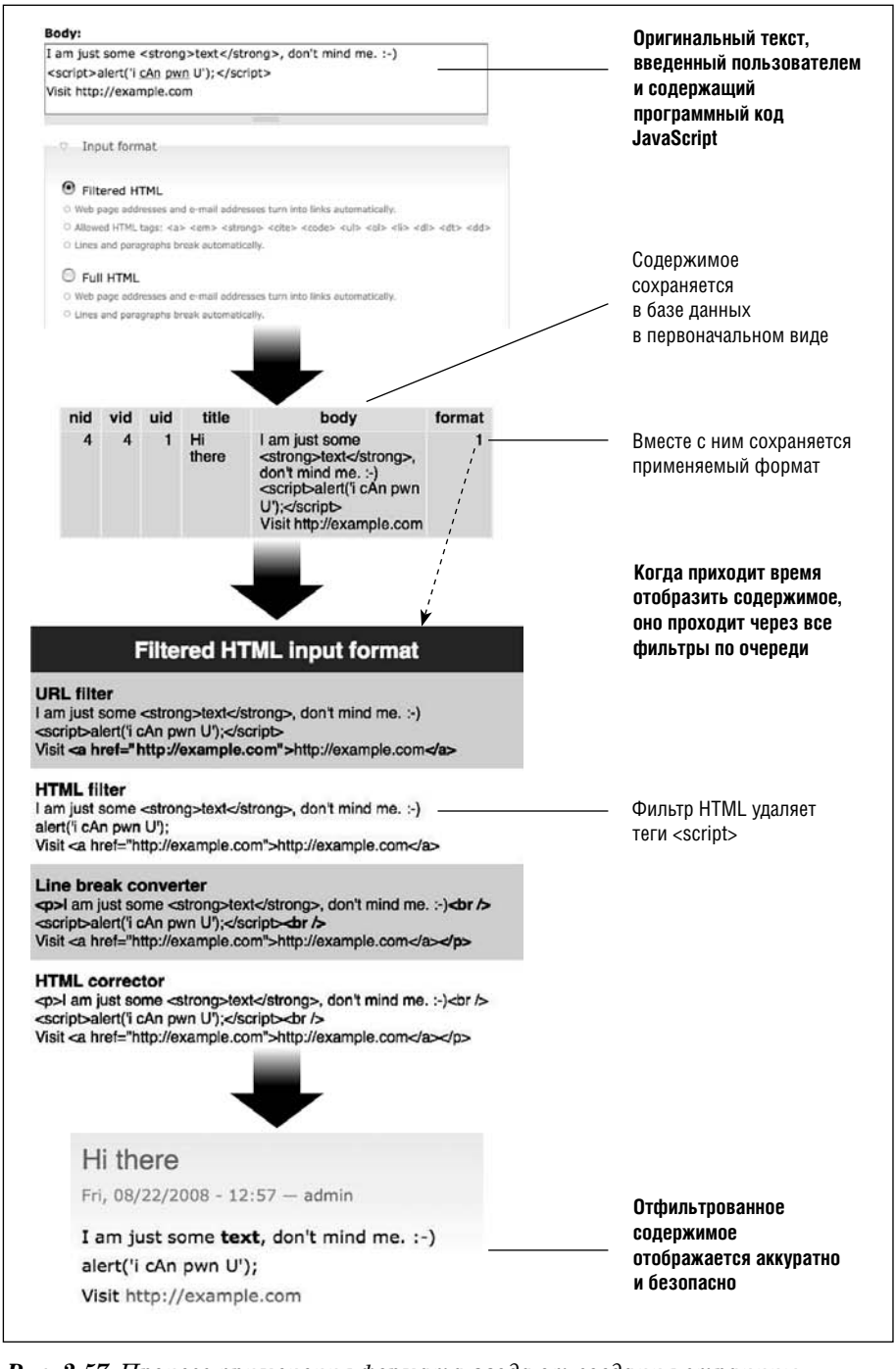


Рис. 2.57. *Процесс применения формата ввода от создания страницы до отображения*

Система Drupal выполняет фильтрацию только при выводе содержимого, поэтому, если в содержимом присутствует что-то недопустимое, его по-прежнему можно будет наблюдать при редактировании – недопустимые фрагменты удаляются только при отображении. Если вы заметите это и подумаете, что что-то не в порядке, проверьте формат ввода, применяемый к содержимому, и убедитесь, что выбран формат, который не отфильтровывает то, что вы хотели бы отобразить на экране. Обычно такое часто случается при попытке отобразить изображение с использованием формата ввода Filtered HTML (Фильтрованный HTML). Вы можете расширить перечень тегов, которые будет пропускать фильтр HTML, на странице Administer (Администрирование)→Site configuration (Настройка сайта)→Input formats (Форматы ввода) (*admin/settings/filters*), приведя его в соответствие со своими требованиями.

Практика: настройка модуля FCKeditor

Прежде чем перейти к настройкам редактора, нам требуется установить все необходимые модули. Загрузите последние версии модулей FCKeditor и IMCE modules, как мы делали это раньше с модулем Administration Menu, в разделе «Практика: работа с модулями». Страница проекта FCKeditor находится по адресу *http://drupal.org/project/fckeditor*, а страница проекта IMCE – по адресу *http://drupal.org/project/imce*. Однако просто установить модули будет недостаточно, потому что в действительности модуль FCKeditor не содержит реализацию самого редактора FCKeditor. Это обусловлено тем, что открытый проект редактора FCKeditor располагается на своем собственном веб-сайте, а его сообщество разработчиков и пользователей не является частью сообщества Drupal.

1. Откройте страницу *http://www.fckeditor.net* и щелкните на ссылке Download (Загрузить) в области навигации по сайту, расположенной в правом верхнем углу, как показано на рис. 2.58.
2. Первой в списке, в разделе Download (Загрузить) будет самая последняя версия редактора FCKeditor. На рис. 2.59 – это версия FCKeditor 2.6.3. Под названием версии имеются две ссылки, которые можно использовать для загрузки. Единственное отличие между ними – тип сжатия архива, zip или tar; вы можете воспользоваться любой из них. Щелкните на желаемой ссылке под требуемой версией. На рис. 2.59 это может быть *FCKeditor_2.6.3.zip* или *FCKeditor_2.6.3.tar.gz*.
3. После загрузки файла его необходимо распаковать. В результате вы должны получить папку с именем *fckeditor*.
4. Скопируйте эту папку в каталог модуля FCKeditor. Там уже существует пустая папка с таким названием, поэтому вам необходимо *заменить* эту папку новой, только что извлеченной из архива. Из-за

похожих имен может возникнуть путаница, поэтому убедитесь, что в результате этих манипуляций вы получили следующую структуру каталогов, где должен находиться основной файл на языке PHP с реализацией редактора FCKeditor (которая также показана на рис. 2.60): *sites/all/modules/fckeditor/fckeditor/fckeditor.php*.

Рис. 2.58. *Ссылка Download на веб-сайте FCKeditor*

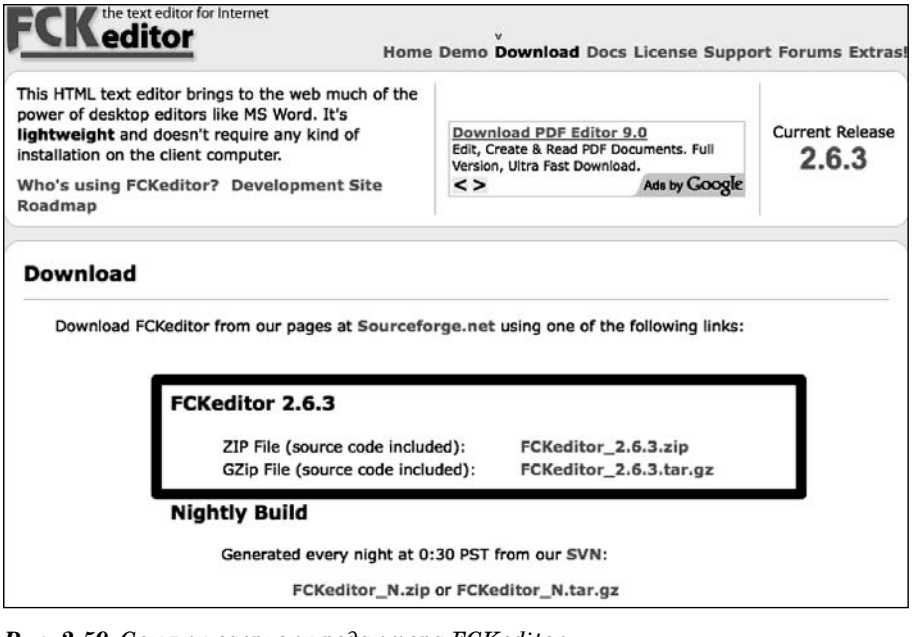


Рис. 2.59. *Ссылки загрузки редактора FCKeditor*

Теперь, когда у нас имеются все необходимые модули и сам редактор FCKeditor, можно продолжить установку и настройку.

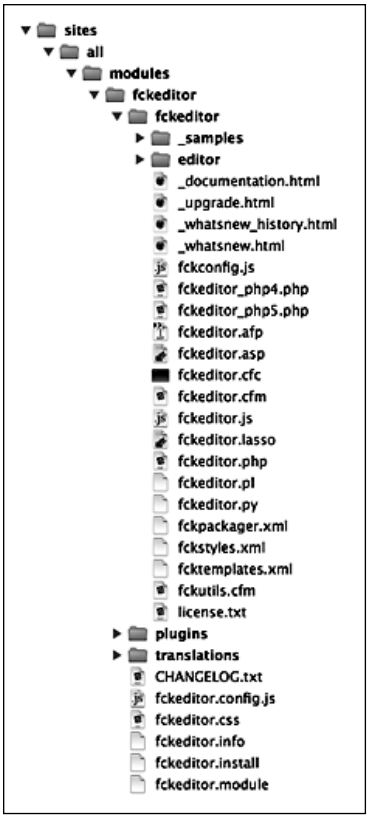


Рис. 2.60. *Структура каталогов на веб-сайте для модуля FCKeditor*

1. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Modules (Модули) (*admin/build/modules*) и включите следующие модули:
   - Other (Другой)
     – FCKeditor
     – IMCE
2. Перейдите на страницу Administer (Администрирование)→User management (Управление пользователями)→Permissions (Разрешения) (*admin/user/permissions*) и установите разрешения в соответствии с табл. 2.22. По окончании щелкните на кнопке Save permissions (Сохранить права доступа).

Таблица 2.22. Разрешения для модуля FCKeditor

Разрешение: модуль fckeditor	**anonymous user (анонимный пользователь)**	**authenticated user (зарегистрированный пользователь)**	**editor (редактор)**	**site administrator (администратор сайта)**
access fckeditor (доступ к fckeditor)	отмечен	отмечен	отмечен	отмечен
administer editor (управлять редактором)				отмечен
allow fckeditor file uploads (разрешить модулю fckeditor выгружать файлы)			отмечен	отмечен

В обычной ситуации было бы достаточно включить разрешения для анонимных и зарегистрированных пользователей, так как все остальные роли уже обладают разрешениями зарегистрированных пользователей. Однако профили редактора FCKeditor опираются на наличие в ролях явно установленного разрешения «access fckeditor».

3. Изначально пользователю №1 не присвоена ни одна из ролей, кроме роли зарегистрированного пользователя, поэтому щелкните на ссылке My account (Мой профиль), затем на вкладке Edit (Изменить) (*user/1/edit*) и включите роль site administrator (администратор сайта).
4. Перейдите на страницу Administer (Администрирование)→Site configuration (Настройка сайта)→FCKeditor (*admin/settings/fckeditor*), где находятся настройки редактора FCKeditor.
5. По умолчанию существует два профиля, но они не вполне соответствуют нашим требованиям. Давайте перенацелим их и изменим несколько параметров в соответствии с нашими потребностями. Щелкните на ссылке edit (изменить) в профиле Default (По умолчанию) и внесите изменения в соответствии с табл. 2.23. По окончании щелкните на кнопке Update profile (Обновить профиль), чтобы сохранить новый профиль. Этот шаг обеспечит доступ к некоторым простым средствам форматирования для большинства пользователей.

Таблица 2.23. Настройки профиля Default (по умолчанию) редактора FCKeditor

Параметр	Значение
Basic setup (Базовые настройки)	
Roles allowed to use this profile (Роли, допустимые для использования в этом профиле)	anonymous user (анонимный пользователь) authenticated user (зарегистрированный пользователь)
File browser settings (Параметры выбора файлов)	
File browser type (Инструмент выбора файлов)	IMCE

6. Вернемся на страницу с настройками FCkeditor Administer (Администрирование)→Site configuration (Настройка сайта)→FCKeditor (*admin/settings/fckeditor*). Нашим редакторам, Майку и Дженни, для создания содержимого потребуются более широкие возможности форматирования, поэтому для них и для Голди мы задействуем профиль Advanced (Продвинутый). Щелкните на ссылке edit (изменить) рядом с профилем Advanced (Продвинутый) и установите настройки в соответствии с табл. 2.24. По окончании щелкните на кнопке Update profile (Обновить профиль).

Таблица 2.24. Настройки профиля Advanced редактора FCKeditor

Параметр	Значение
Basic setup (Базовые настройки)	
Roles allowed to use this profile (Роли, допустимые для использования в этом профиле)	editor site administrator (Обратите внимание: вы должны снять флажок для роли authenticated user (зарегистрированный пользователь))
File browser settings (Параметры выбора файлов)	
File browser type (Инструмент выбора файлов)	IMCE

Редактор FCKeditor предоставляет возможность выбора для настройки панелей инструментов. Если появится необходимость определить, какие кнопки будут отображаться на панели инструментов, или создать новую панель инструментов, это можно будет сделать, изменив конфигурационный файл, входящий в состав модуля. Файл *fckeditor.config.js* находится внутри каталога *sites/modules/fckeditor*. Открыв его, можно увидеть определения панелей инструментов с кнопками, перечисленными под ними. Прежде чем вносить какие-либо изменения, обязательно скопируйте оригинальную версию файла, чтобы всегда можно было вернуться к нему, когда возникнет такая необходимость.

Теперь в нашем распоряжении имеется несколько панелей инструментов, но есть одна небольшая проблема, связанная с форматами ввода. Редактор FCKeditor будет добавлять в содержимое разметку HTML, но Drupal, используя формат по умолчанию Filtered HTML (Фильтрованный HTML), удалит большую часть тегов. Мы используем две панели инструментов, удовлетворяющих разным потребностям.

На панели инструментов для роли «editor» имеется множество кнопок, которые предполагают использование тегов HTML всех видов. Кроме того, эта панель будет доступна только самым доверенным пользователям: администратору сайта и владельцам. Поскольку эта панель инструментов будет доступна только доверенным пользователям, проще всего будет дать им доступ к формату Full HTML (Фильтрованный HTML), чтобы обеспечить возможность использовать все, что они пожелают, для придания их содержимому привлекательного внешнего вида.

Панель инструментов для обычных и анонимных пользователей предлагает лишь несколько возможных операций, и эти пользователи определенно не пользуются доверием. Нам требуется позволить им использовать чуть больше тегов HTML, чем пропускает формат Filtered HTML (Фильтрованный HTML) по умолчанию, но нам определенно не следует давать им доступ к формату Full HTML (HTML без ограничений). Чтобы обеспечить нормальную работу этой панели инструментов, нам необходимо изменить формат Filtered HTML (Фильтрованный HTML), чтобы он пропускал необходимые дополнительные теги.

1. Перейдите на страницу Administer (Администрирование)→Site configuration (Настройка сайта) →Input formats (Форматы ввода) (*admin/settings/filters*), где перечислены наши два формата по умолчанию.
2. Щелкните на ссылке configure (настроить) для формата Full HTML (Фильтрованный HTML) (*admin/settings/filters/2*).
3. Вы увидите список ролей, которые могут использовать этот формат. В настоящее время ни одна роль не выбрана. Отметьте флажки ролей editor и site administrator и затем щелкните на кнопке Save configuration (Сохранить конфигурацию).

4. Теперь профиль Advanced (продвинутый) будет работать, как требуется. Теперь обратим наше внимание на профиль Default (По умолчанию). Щелкните на ссылке configure (настроить) для формата Filtered HTML (Фильтрованный HTML) (*admin/settings/filters/1*).
5. Чтобы изменить фильтр HTML, который определяет, какие теги можно использовать, а какие нет, щелкните на вкладке Configure (Настроить) (*admin/settings/filters/1/configure*).
6. В разделе HTML filter (Фильтр HTML) вы должны увидеть список тегов HTML в поле Allowed HTML tags (Допускаются только следующие теги HTML). В конец этого списка нам необходимо добавить следующие теги: `<img>` `<div>` `<pre>` `<address>` `<h1>` `<h2>` `<h3>` `<h4>` `<h5>` `<h6>` (обратите внимание на пробелы между тегами). Эти теги позволят использовать кнопку вставки изображений и корректно отображать список форматов в раскрывающемся меню Format (Формат). По окончании щелкните на кнопке Save configuration (Сохранить конфигурацию).

Теперь у нас имеется прекрасная панель инструментов для пользователей, что позволит им форматировать свои сообщения с использованием форматов Drupal, как показано на рис. 2.61.

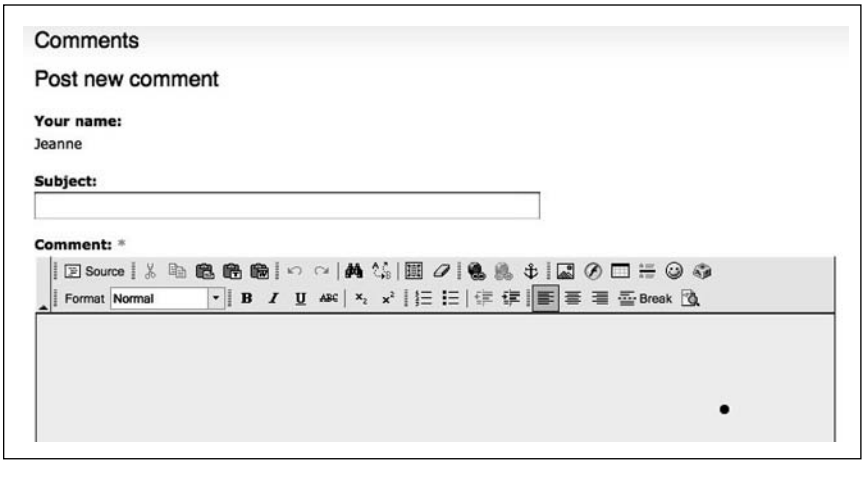


Рис. 2.61. *Улучшенная панель инструментов, доступная пользователям с ролями editor и site administrator*

Однако если немного поэкспериментировать с панелью инструментов, можно заметить, что при использовании кнопки вставки изображения она требует указать адрес URL изображения. Но Майк и Дженни с этим вряд ли разберутся. Им необходимо предоставить возможность отыскать изображение на своем компьютере, выгрузить его на сайт и вставить. Чтобы кнопка вставки изображения могла выгружать изображения на сайт, следует настроить модуль IMCE.

1. Перейдите на страницу Administer (Администрирование)→Site configuration (Настройка сайта)→IMCE (*admin/settings/imce*).
2. Здесь можно видеть два предустановленных профиля. Профиль User-1 очевидно предназначен для пользователя №1, но он прекрасно отвечает потребностям редакторов и администратора сайта. Для обычных пользователей можно задействовать профиль Sample (Пример). Щелкните на ссылке Edit (Изменить) для профиля User-1.
3. Здесь нам нужно изменить несколько настроек. Измените значение поля Profile name (Имя профиля) на Advanced, чтобы оно соответствовало названию профиля FCKeditor. Это будет показывать нам, что они связаны между собой в терминах настройки сайта, но для обеспечения нормальной работы модуля этого не требуется.
4. Кроме того, следует изменить значение поля Maximum image resolution size (Максимальный размер изображения) на 800x600, так как этого вполне достаточно для большинства случаев, а размер 1200×1200 может привести к нарушениям в размещении элементов страниц на нашем сайте.
5. Щелкните на кнопке Save configuration (Сохранить конфигурацию).
6. Чтобы сохранить непротиворечивость имен, переименуем наш профиль. Вернитесь на страницу Administer (Администрирование)→Site configuration (Настройка сайта)→IMCE (*admin/settings/imce*) и щелкните на ссылке Edit (Изменить) для профиля Sample (Пример).
7. Измените значение поля Profile name (Имя профиля) на Default. Остальные настройки со значениями по умолчанию вполне устраивают нас, поэтому просто щелкните на кнопке Save configuration (Сохранить конфигурацию).
8. Последнее, что нужно сделать, – это убедиться в корректном соответствии профилей и ролей. Вернитесь на страницу Administer (Администрирование)→Site configuration (Настройка сайта)→IMCE (*admin/settings/imce*) и в разделе Role-profile assignments (Назначения профилей ролям) выполните присваивания в соответствии с табл. 2.25. (Обратите внимание, что при этом не следует изменять веса.) Мы не задействовали ни один из профилей для анонимных пользователей, поэтому они смогут указывать ссылки на изображения, но не выгружать их на сервер компании «Mom and Pop».

Таблица 2.25. Соответствие ролей и профилей для модуля IMCE

Роль	**Присвоенный профиль**
site administrator	Advanced
editor	Advanced
authenticated user (зарегистрированный пользователь)	Default

9. Щелкните на кнопке Save configuration (Сохранить конфигурацию), и мы закончили!

Если теперь перейти на страницу создания нового содержимого, вы увидите отличную панель инструментов, как показано на рис. 2.61.

В заключение

В этой главе был представлен обзор основных функциональных возможностей системы Drupal по созданию небольших веб-сайтов. Мы установили Drupal, добавили новые типы содержимого и наполнили сайт некоторым содержимым, мы получили представление о системе таксономии, создали форму контактов, узнали, как настраиваются темы оформления Drupal, чтобы придать сайту определенный внешний вид, и снабдили наших редакторов визуальным редактором и средством для работы с изображениями, чтобы они могли придать своим материалам желаемый внешний вид, не обладая при этом знанием языка разметки HTML.

Ниже приводится список модулей, использованных в этой главе:

- Administration menu: *http://drupal.org/project/admin_menu*
- FCKeditor: *http://drupal.org/project/fckeditor*
- IMCE: *http://drupal.org/project/imce*

Ниже приводится список ссылок, упоминавшихся в этой главе:

- Страница для загрузки ядра Drupal: *http://drupal.org/download*
- Страница для загрузки модулей: *http://drupal.org/project/Modules*
- Страница для загрузки тем оформления: *http://drupal.org/project/Themes*
- Веб-сайт проекта FCKeditor: *http://fckeditor.net*
- Системные требования Drupal: *http://drupal.org/node/270*
- Сведения о версиях Drupal: *http://drupal.org/handbook/version-info*
- Руководства на сайте Drupal.org: *http://drupal.org/handbooks*
- Руководство для начинающих: *http://drupal.org/node/258*
- Сборник часто задаваемых вопросов по устранению проблем: *http://drupal.org/Troubleshooting-FAQ*
- Форум поддержки: *http://drupal.org/forum/18*
- Локальный веб-сервер XAMPP: *http://www.apachefriends.org/en/xampp.html*
- Локальный веб-сервер MAMP: *http://mamp.info/en/download.html*
- Программа StuffIt Expander: *http://my.smithmicro.com/mac/stuffit-expander*

3

Каталог вакансий

Эта глава представляет две наиболее важных особенности Drupal. Да, мы прямо говорим, что две наиболее важные особенности – это пакет Content Construction Kit и модуль Views. Пакет *Content Construction Kit* (часто для краткости обозначается, как CCK) позволяет, не написав ни строчки кода, создавать формы с разнообразными полями, такими как флажки, раскрывающиеся списки, поля выгрузки изображений и многие другие. Модуль *Views* является естественным дополнением к пакету CCK, позволяя получать данные с сайта. Модуль Views дает возможность создавать страницы и блоки, которые получают данные и отображают их перед посетителями. Хотите отобразить таблицу с описаниями продуктов, разбитую на страницы и допускающую возможность сортировки по цене и названию производителя? Вы можете создать ее с помощью пакета CCK и модуля Views. Хотите отобразить блок со списком альбомов определенного музыканта в виде мозаики, состоящей из изображений обложек? Вы можете создать его с помощью пакета CCK и модуля Views. Какой бы список содержимого на вашем веб-сайте вы не взяли (а большинство веб-сайтов практически *всегда* всего лишь перечисляют содержимое в той или иной форме), CCK и Views окажутся теми ключевыми модулями, которые вам необходимы.

Пакет CCK и модуль Views являются фундаментом почти всех проектов, рассматриваемых в этой книге, и большинства веб-сайтов в Интернете, работающих под управлением системы Drupal. Мы рассмотрим, как создать новый тип содержимого и как настроить форму, чтобы вы могли добавлять в нее поля для ввода данных любых типов. Мы настроим сайт, позволяющий создавать объявления о вакансиях, и сконструируем интерфейс для просмотра доступных вакансий.

В этой главе будут представлены следующие модули:

CCK (http://drupal.org/project/cck)

Добавляет новые поля в типы содержимого.

Views (http://drupal.org/project/views)

Создает списки содержимого и пользователей.

FileField (http://drupal.org/project/filefield)

Добавляет в типы содержимого поля, предназначенные для выгрузки файлов.

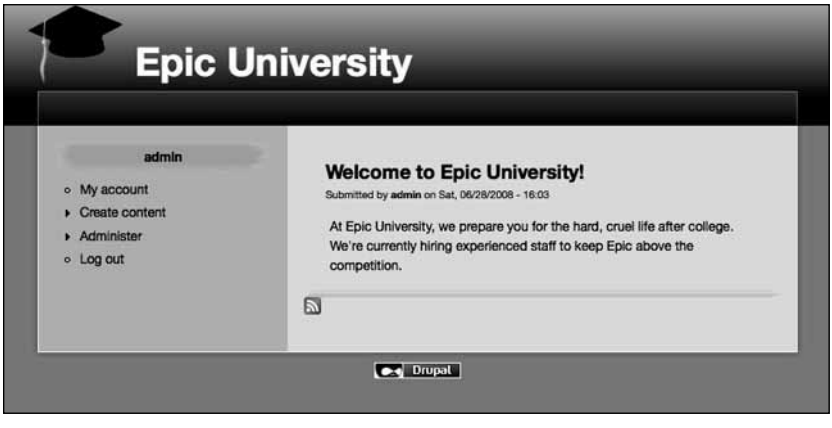


Рис. 3.1. *Сайт Epic University сразу после установки*

Если вы собираетесь следовать за практическими упражнениями этой главы, вам следует установить Drupal, используя профиль установки Job Posting из примеров кода, распространяемых вместе с книгой. В результате применения этого профиля на вашем веб-сервере будет создана заготовка веб-сайта. В окончательном виде этот веб-сайт будет выглядеть, как показано на рис. 3.1 и по адресу: *http://jobs.usingdrupal.com*. За дополнительной информацией об использовании профилей установки обращайтесь к разделу «Вступление».

Задача

Нескольким студентам университета Epic University в качестве исследовательской работы было задано создать веб-сайт каталога вакансий для их учебного заведения. Университету необходимо в самые кратчайшие сроки получить сайт, работающий на его внутренних серверах. Так как конструированием сайта занимаются студенты, очевидно, что бюджетные затраты на его создание весьма ограничены. Так как сайт должен обладать гибкой системой узлов, возможностью управления пользователями и низкой стоимостью, студенты решили загрузить систему Drupal и приступить к созданию сайта на ее основе.

Отделу кадров необходимо, чтобы факультеты университета могли размещать на сайте объявления о вакансиях, включая описание, название факультета, имя ответственного лица и информацию о заработной пла-

те. Пользователи должны иметь возможность регистрироваться на сайте и просматривать списки имеющихся вакансий по факультетам. Кроме того, пользователи должны иметь возможность отправлять заявки по определенным позициям и просматривать результаты по всем позициям, на которые они отправляли свои заявки.

Примечания по реализации

Ядро системы Drupal является неплохой отправной точкой для создания этого сайта. Оно обеспечивает необходимую идентификацию пользователей и позволяет создавать различные типы содержимого, такие как «вакансия» и «заявка». Функциональные возможности Drupal изначально обеспечивают эти новые типы содержимого только полями «Title» (Заголовок) «Body» (Содержание). Нам потребуется еще несколько полей, чтобы пользователи могли вводить данные в разные поля и чтобы мы могли извлекать информацию из определенных полей при создании списков содержимого.

Нестандартные формы ввода

В основе требований, предъявляемых к этому веб-сайту, – ввод текстовых данных, которые будут добавляться с помощью различных форм. Пакет CCK обеспечивает необходимые средства ввода данных. С его помощью мы сможем добавить несколько необходимых полей различных типов, таких как раскрывающиеся списки с возможностью выбора факультета университета и простые текстовые поля для ввода номера телефона и адреса.

Отображение содержимого

Помимо ввода данных соискатели вакансий и работодатели должны иметь возможность просматривать списки потенциальных вакансий и претендентов.

Модуль Views в состоянии удовлетворить практически любым целям отображения содержимого и позволяет выводить содержимое различными способами: в виде таблиц, в виде списков с полным содержимым узлов или с анонсами, в виде ленты RSS, в виде списка отдельных полей и множеством других способов. В этой главе мы будем создавать все списки в виде представлений, включая специализированные представления, которые принимают идентификационный номер пользователя или название факультета и выполняют фильтрацию, отображая только соответствующее содержимое.

Выгрузка файлов

Ядро системы Drupal включает модуль Upload, позволяющий прикреплять файлы к содержимому, однако он обладает некоторыми недостатками, перечисленными ниже:

- По умолчанию возможность выгрузки файлов включена для всех типов содержимого. Однако в большинстве случаев требуется обеспечить такую возможность только для одного-двух типов содержимого.
- Все файлы выгружаются в «корневой» каталог, предназначенный для ваших файлов, что может привести к путанице, когда в нем накопится несколько сотен файлов.
- Модуль Upload позволяет выгружать на сервер неограниченное число файлов для каждого узла. Но иногда бывает необходимо, чтобы пользователи имели возможность выгружать только один файл для каждого узла или не более трех (к примеру).
- Чтобы обеспечить возможность выгружать файлы с помощью модуля Upload, необходимо дополнить набор параметров File attachments (Выгрузка файлов), однако нередко файл является неотъемлемой частью содержимого, поэтому такое «углубление» в настройки нежелательно.
- Размеры квот и списки допустимых расширений имен файлов распространяются на весь сайт для всех пользователей, наделенных определенной ролью, однако зачастую требуется более гибкий подход в определении этих типов и настроек.
- Модуль Upload не позволяет иметь более одного поля выбора файла или давать ему какое-то определенное имя.

К счастью для нас, сторонний модуль FileField для CCK помогает обойти все эти недостатки. Поля для выгрузки файла могут добавляться и настраиваться отдельно для каждого типа данных и позволяют выполнять более тонкую настройку.

В центре внимания: CCK

Пакет CCK обеспечивает чрезвычайно гибкую платформу создания форм ввода содержимого. Ядро Drupal дает возможность определять различные типы содержимого (такие как «Job Opening» (Открытая вакансия) или «Application» (Заявка)), но в нем отсутствует механизм, позволяющий добавлять поля к вновь создаваемым типам. Пока пакет CCK не станет частью ядра Drupal (что ожидается в будущих версиях), первым шагом на пути создания почти любого сайта на базе Drupal должна стать установка CCK.

Сразу после установки новой копии Drupal имеется два типа содержимого: Story (Статья) и Page (Страница). По сути оба типа представляют собой одно и то же – форму, содержащую поле Title (Заголовок) и поле Body (Содержание). Любые другие создаваемые типы содержимого также будут иметь поле Title (Заголовок) и (необязательно) поле Body (Содержание), а с помощью дополнительного модуля, такого как CCK, можно добавлять к типам содержимого дополнительные поля. На

Create Story

Title: *

Split summary at cursor

Body:

Web page addresses and e-mail addresses turn into links automatically.

Allowed HTML tags: <a>   <cite> <code>   <dl> <dt> <dd>

Lines and paragraphs break automatically.

More information about formatting options

Save Preview

Рис. 3.2. *Форма создания материала типа Story (Статья) в том виде, в каком она выводится для зарегистрированных пользователей на новом сайте Drupal*

Create Story

Title: *

By line:

Split summary at cursor

Body:

Web page addresses and e-mail addresses turn into links automatically.

Allowed HTML tags: <a>   <cite> <code>   <dl> <dt> <dd>

Lines and paragraphs break automatically.

More information about formatting options

Image:

Browse...

Upload

Options:

N/A

Sample choice 1

Sample choice 2

Sample choice 3

Save Preview

Рис. 3.3. *Форма ввода содержимого типа Story (Статья) после добавления нестандартных полей*

рис. 3.2 изображен тип содержимого Story до того, как к нему с помощью пакета CCK будут добавлены дополнительные поля.

После включения пакета CCK появляется возможность добавлять любые поля к любому типу содержимого. В комбинации с возможностью ядра Drupal создавать любое число нестандартных типов содержимого пакет CCK позволяет создавать любое число нестандартных форм добавления содержимого. На рис. 3.3 показана та же форма после добавления нескольких дополнительных полей, таких как текстовое поле, поле с изображением и группа радиокнопок.

CCK или Taxonomy?

Оба модуля, Taxonomy и CCK, позволяют добавлять новые элементы в формы создания содержимого. Ниже приводится несколько правил, которыми можно руководствоваться при выборе того или иного способа:

- Основная цель модуля Taxonomy – создание категорий, поэтому, если вам требуются категории, следует использовать модуль Taxonomy. Прежде чем с помощью пакета CCK создавать поле с именем «Category» (Категория), дважды подумайте. Специально для этой цели был создан модуль Taxonomy, и многие существующие модули обеспечивают возможность прямой интеграции с модулем Taxonomy.
- Применение таксономии обеспечивает возможность иерархической классификации, которую легко можно создать и организовать. Если ваши категории требуется представлять в виде древовидной структуры, использование модуля Taxonomy будет более оправданно.
- Модуль Taxonomy предоставляет для категорий только поля «Title» (Заголовок) и «Description» (Описание). Если категория должна хранить больше информации, то лучше будет создать новый тип содержимого для представления категории, а затем с помощью пакета CCK создать поле ссылки на узел для выбора узлов, принадлежащих этой категории.
- И главное правило: если имеется вероятность, что поле может быть удалено, но важно не утратить при этом тип содержимого, то лучше использовать модуль Taxonomy. Статья, отнесенная к категории «Технологии», так и останется статьей, даже если вы удалите связь с категорией, и в таких случаях модуль Taxonomy лучше отвечает вашим потребностям. Но если поле является частью элемента содержимого, например имя музыканта, записавшего альбом, то тогда лучшим выбором будет пакет CCK.

После добавления полей пакет CCK автоматически обеспечивает сохранение информации в базе данных, а после отправки данных пользователями обеспечивает возможность представления этой информации разными способами.

Поля

Функционально для конечного пользователя пакет CCK состоит из двух частей. Первая из них – это *поля*, которые дают пользователю возможность сохранять свои данные на сайте. Поля определяют типы сохраняемых данных, такие как целое число, числа с фиксированной десятичной точкой или текст. При добавлении нового поля к типу содержимого в первую очередь необходимо определить, какой тип данных будет хранить форма «за кулисами». Будет ли информация вводиться как обычное число или как простой текст, или она будет чем-то более сложным, например ссылкой на другой узел или на пользователя? Типы полей, включенные в «ядро» CCK, перечислены в табл. 3.1. Другие модули, такие как Fivestar (*http://drupal.org/project/fivestar*), ImageField (*http://drupal.org/project/imagefield*) и Date (*http://drupal.org/project/date*), добавляют свои типы полей в пакет CCK. Эти модули, реализующие новые типы полей для CCK, рассматриваются ниже в этой книге, в главах 4, 7 и 9 соответственно. Полный список всех доступных типов полей для CCK можно найти по адресу: *http://drupal.org/taxonomy/term/88*.

Таблица 3.1. Встроенные типы полей в пакете CCK

Тип поля	Назначение
Integer	Наиболее эффективный способ хранения чисел. Используется для обозначения номеров, идентификаторов и любой другой информации, имеющей числовое выражение, например номеров дорожек в альбоме или количества участников некоторого события.
Decimal	Эффективный способ хранения чисел с фиксированной десятичной точкой. Удобно использовать для представления денежных сумм.
Float	Самый лучший способ хранения чисел, когда требуется высокая точность, например результаты научных измерений.
Text	Может сохранять произвольные текстовые строки. Удобно использовать для представления названий и описаний, а также для сохранения длинных фрагментов текста, таких как биография.
Node Reference	Поля этого типа могут ссылаться на любой узел сайта. Удобно использовать, когда необходимо связать одну часть содержимого с другой.

User Reference	Поля этого типа могут ссылаться на любого пользователя сайта. Удобно, когда необходимо связать пользователя с определенной частью содержимого, например обозначить координатора или ответственное лицо некоторого события.

Виджеты

Как только тип данных будет определен, наступает время подумать, как это поле будет выглядеть в форме. На языке ССК элементы форм называются виджетами. Вам требуется раскрывающийся список или группа радиокнопок? Флажки или текстовые поля с функцией автодополнения? Выберите виджет, который лучше всего подойдет для ввода данных пользователем. Следует отметить, что перечень доступных виджетов зависит от типа поля.

Типы виджетов, включенные в ядро ССК, перечислены в табл. 3.2. Как и в случае с полями, часто дополнительные модули расширяют перечень доступных виджетов.

Таблица 3.2. Встроенные типы виджетов в пакете ССК

Тип виджета	Назначение
Текстовое поле	Этот виджет позволяет вводить единственную строку текста, например имя или номер телефона. Поддерживается как простой ввод, так и форматированный.
Текстовая область (многострочный текст)	Этот виджет может использоваться для ввода больших параграфов текста, таких как биография или описание продукта. Поддерживается как простой ввод, так и форматированный.
Одиночный флажок	Используется для представления информации, которая может быть выражена словами «да» или «нет», например поле, которое запрашивает, желает ли пользователь подписаться на рассылку по электронной почте.
Группа флажков или радиокнопок	Используется, когда необходимо предоставить возможность выбора из нескольких вариантов; флажки используются, когда допускается выбор нескольких вариантов одновременно, а радиокнопки – когда допускается выбор единственного значения. В большинстве случаев поле с названием «Пол» имеет смысл представить как группу радиокнопок, тогда как поле «Предпочтительный цвет» – в виде группы флажков.

Таблица 3.2 (продолжение)

Тип виджета	Назначение
Раскрывающийся список	Альтернативой группе флажков или радиокнопок является раскрывающийся список. Удобно использовать, когда для выбора предоставляется множество различных значений, которые сложно было бы представить в виде отдельных элементов выбора.
Текстовое поле с функцией автодополнения	Этот виджет отображает текстовое поле, которое по мере ввода отображает список возможных вариантов, начинающихся с введенной последовательности символов. Например, после ввода символов «st» поле могло бы предложить варианты «Stewart», «Studebaker» и «Style». Удобно использовать, когда количество вариантов настолько огромно, что отображение их в виде раскрывающегося списка было бы затруднительно. Однако это поле требует, чтобы пользователь имел представление о том, что он ищет.

Форматтеры

В дополнение к настройке поля ввода форматтеры позволяют указать, как будут выглядеть данные при отображении на страницах сайта. Например, десятичные числа могут отображаться с запятой, отделяющей группы разрядов, или без нее, как показано на рис. 3.4.

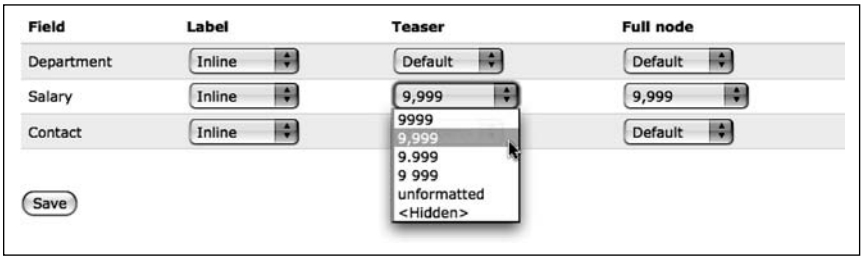


Рис. 3.4. *Настройка формата отображения поля*

Другие модули могут добавлять дополнительные форматы, обеспечивая множество способов представления информации. Модуль ImageCache, о котором рассказывается в главе 7, является одним из примеров таких модулей – он позволяет изменять размеры отображаемых изображений.

Имейте в виду, что перечень доступных форматов зависит от типа поля, поэтому очень важно определить тип поля как integer, decimal или float, если вы собираетесь отображать числа. Пакет CCK не позволит изменить тип данных после того, как поле будет настроено, поэтому при не-

обходимости изменить тип поля text на integer (или на любой другой тип) нужно удалить поле и создать новое с требуемым типом.

Практика: CCK

Прежде чем приступить к созданию нашего веб-сайта каталога вакансий, давайте подумаем, какие типы содержимого нам потребуются для обеспечения всех функциональных возможностей. Для сайта потребуется два типа содержимого:

Job Type (вакансия)

Описание конкретной вакансии.

Job Application Type (заявка)

Заявка, связывающая соискателя и вакансию.

Нам потребуется связывать заявки с соответствующими вакансиями, а вакансии – с соответствующими контактными лицами. Эту возможность обеспечат поля типа node reference (ссылка на узел) и user reference (ссылка на пользователя), упоминавшиеся выше в этой главе.

Приступая к конструированию новых типов содержимого в системе Drupal, лучше всего начинать с того, чтобы представить себе, как будет выглядеть форма, которую вы пытаетесь создать. На рис. 3.5 показаны макеты обеих создаваемых нами форм – формы ввода информации о вакансии и формы ввода заявки.

1. Чтобы включить пакет CCK, перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Modules (Модули) (*admin/build/modules*). Включите две части, имеющие отношение к пакету CCK: модуль Content, который является «ядром» CCK, и один или более модулей, представляющих «поля» и «виджеты».

Модуль FileField – это самостоятельный модуль, который не включается в пакет CCK по умолчанию. Его можно загрузить со страницы *http://drupal.org/project/filefield*, а кроме того, он поставляется в составе исходных кодов примеров к книге.

2. Включите следующие модули и щелкните на кнопке Save configuration (Сохранить конфигурацию). После включения модулей страница должна выглядеть, как показано на рис. 3.6:
   - CCK:
     - Content
     - FileField
     - Node Reference
     - Number
     - Option Widgets
     - Text
     - User Reference

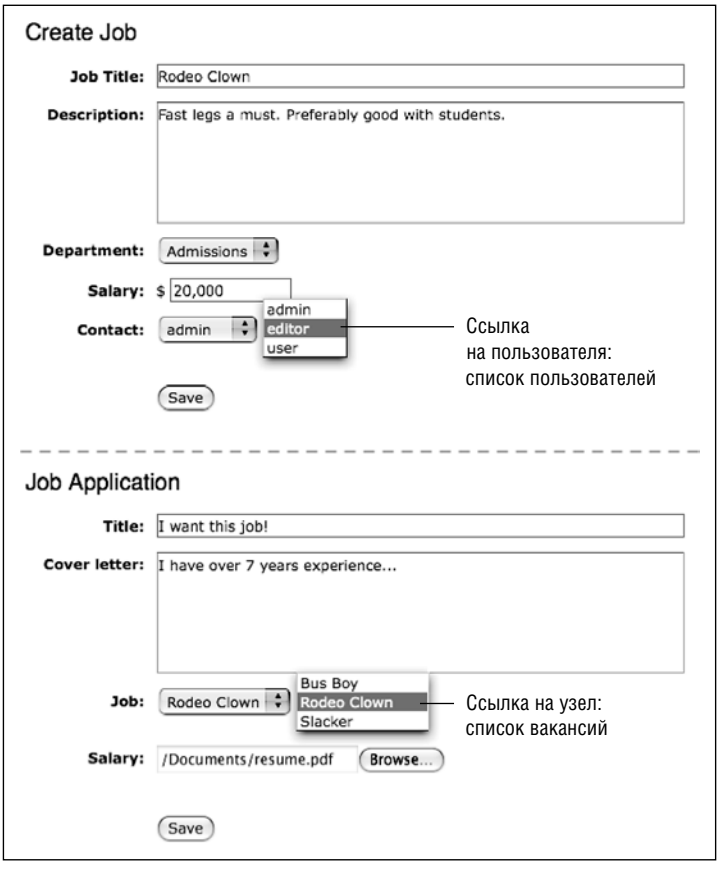


Рис. 3.5. *Макеты форм, необходимых для веб-сайта каталога вакансий*

Практика: тип содержимого Job (вакансия)

Для создания сайта нам потребуются дополнительные типы содержимого, помимо типов Page и Story, предлагаемых ядром Drupal по умолчанию. В этом разделе описывается, как создавать новые типы содержимого и добавлять в них новые поля с помощью пакета CCK.

Содержимое типа Job (Вакансия) будет хранить всю необходимую информацию о конкретной вакансии в университете Epic University. Для этого потребуется определить следующие поля:

- Job Title (Заголовок – обычное поле заголовка узла)
- Description (Описание – обычное поле содержания узла)
- Department (Факультет – текстовое поле с возможностью выбора из списка)

CCK

Enabled	Name	Version	Description
☑	Content	6.x-2.0-beta	Allows administrators to define new content types. Required by: Content Copy (disabled), Content Permissions (disabled), Date (disabled), Date Copy (disabled), Fieldgroup (disabled), FileField (enabled), Node Reference (enabled), Number (enabled), Option Widgets (enabled), Text (enabled), User Reference (enabled), FileField Meta (disabled)
☐	Content Copy	6.x-2.0-beta	Enables ability to import/export field definitions. Depends on: Content (enabled)
☐	Content Permissions	6.x-2.0-beta	Set field-level permissions for CCK fields. Depends on: Content (enabled)
☐	Fieldgroup	6.x-2.0-beta	Create field groups for CCK fields. Depends on: Content (enabled)
☑	FileField	6.x-1.0-beta2	Defines a file field type. Depends on: Content (enabled) Required by: FileField Meta (disabled)
☐	FileField Meta	6.x-1.0-beta2	Add metadata gathering and storage to FileField. Depends on: FileField (enabled), Getid3 (missing), Content (enabled)
☑	Node Reference	6.x-2.0-beta	Defines a field type for referencing one node from another. Depends on: Content (enabled), Text (enabled), Option Widgets (enabled)
☑	Number	6.x-2.0-beta	Defines numeric field types. Depends on: Content (enabled)
☑	Option Widgets	6.x-2.0-beta	Defines selection, check box and radio button widgets for text and numeric fields. Depends on: Content (enabled) Required by: Node Reference (enabled), User Reference (enabled)
☑	Text	6.x-2.0-beta	Defines simple text field types. Depends on: Content (enabled) Required by: Node Reference (enabled), User Reference (enabled)
☑	User Reference	6.x-2.0-beta	Defines a field type for referencing a user from a node. Depends on: Content (enabled), Text (enabled), Option Widgets (enabled)

Рис. 3.6. *Включение пакета CCK*

- Contact (Контакт – поле ссылки на пользователя)
- Salary (Зарплата – целочисленное поле)

Теперь выполним действия по созданию нового типа содержимого:

1. Сначала перейдите на основную страницу настройки типов содержимого Administer (Администрирование)→Content management (Содержание)→Content types (Типы материалов) (*admin/content/types*). Щелкните на вкладке Add content type (Добавить тип содержимого) в верхней части страницы (*admin/content/types/add*).
2. Создайте новый тип содержимого с названием Job, используя настройки из табл. 3.3. Мы изменили назначение полей Title (Заголовок) и Body (Содержание) на Job title (Заголовок вакансии) и Description (Описание) соответственно, просто изменив их метки. Разверните группу параметров Submission form settings (Свойства

формы) и введите заголовки для полей Title и Body. По завершении настройки должны выглядеть, как показано на рис. 3.7.

Таблица 3.3. Настройки типа содержимого Job

Поле	Значение
Identification (Идентификация)	
Name (Имя)	Job
Type (Тип)	job
Description (Описание)	A currently available job position (Вакансия, доступная в настоящее время)
Submission form settings (Свойства формы)	
Title field label (Название поля заголовка)	Job title
Body field label (Название поля текста)	Description

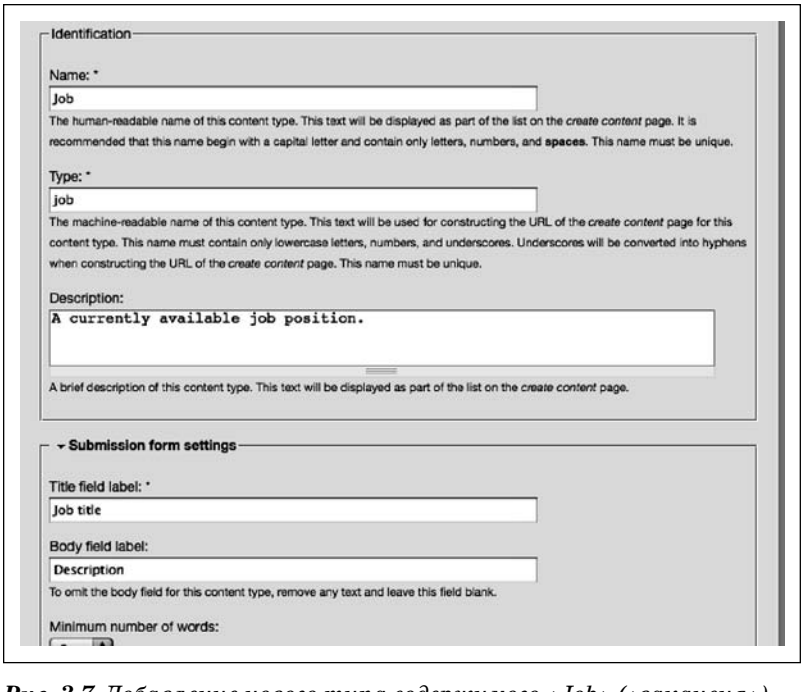


Рис. 3.7. *Добавление нового типа содержимого «Job» («вакансия»)*

3. После сохранения формы щелчком на кнопке Save content type (Сохранить тип содержимого) будет создан новый тип содержимого. Чтобы добавить наше первое поле, щелкните на ссылке manage fields (управлять полями) для типа Job (*admin/content/node-type/job/fields*).
4. Чтобы добавить новый раскрывающийся список для поля Department (Факультет), заполните форму New field (Новое поле), используя настройки из табл. 3.4 и изображенные на рис. 3.8. В качестве типа поля выберите Text (Текст), так как поле будет использоваться для хранения текстовых значений. После выбора типа поля появится список выбора типа виджета.

Таблица 3.4. Значения параметров формы New field (Новое поле) для поля Department

Поле	Значение
Label (Заголовок)	Department
Field name (Имя поля)	department
Select a field type (Выбор типа поля)	Text (Текст)
Select a widget (Выбор виджета)	Select list (Раскрывающийся список)

Стоит потратить пару минут, чтобы хорошо обдумать, какого типа данные будут храниться в поле, прежде чем выбирать тип поля в списке. После того как тип поля будет выбран, его нельзя будет изменить. В случае ошибки вам придется удалить поле и создать новое, указав правильный тип.

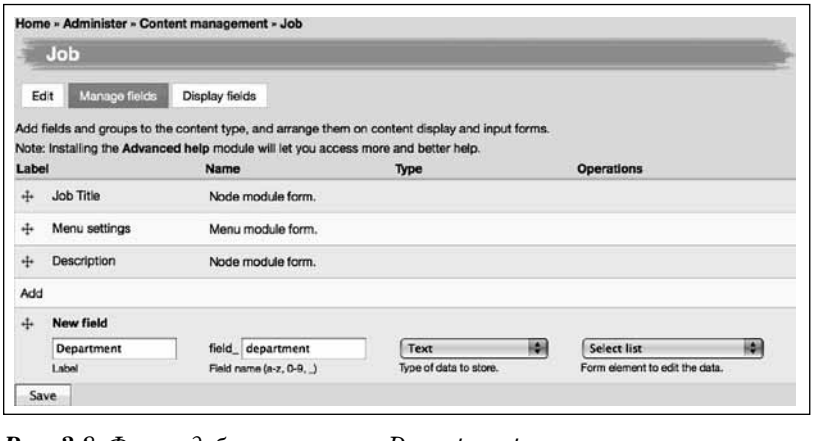


Рис. 3.8. *Форма добавления поля Department*

5. После щелчка на кнопке Save (Сохранить) на следующей странице вы сможете выполнить настройки нового раскрывающегося списка.

Используя значения из табл. 3.5 и изображение на рис. 3.9, настройте раскрывающийся список и заполните его значениями, которые будут доступны пользователям для выбора.

Таблица 3.5. Настройки поля Department

Поле	Значение
Job settings (Настройки представления Job)	
Help text (Справочный текст)	Select the department in which this job belongs. (Выберите факультет, которому принадлежит эта вакансия.)
Global settings (Общие настройки)	
Allowed values list (Список допустимых значений)	Administration (Администрация) Arts (Искусство) Athletics (Спорт) Business (Бизнес) Education (Обучение) Health Sciences (Здравоохранение) Sciences (Наука)

Если вы случайно допустили ошибку в определении поля в форме CCK, ничего страшного. В любой момент можно щелкнуть на ссылке configure (настроить) во вкладке Manage fields (Управлять полями) *(admin/content/node-type/job/fields)* и изменить настройки поля.

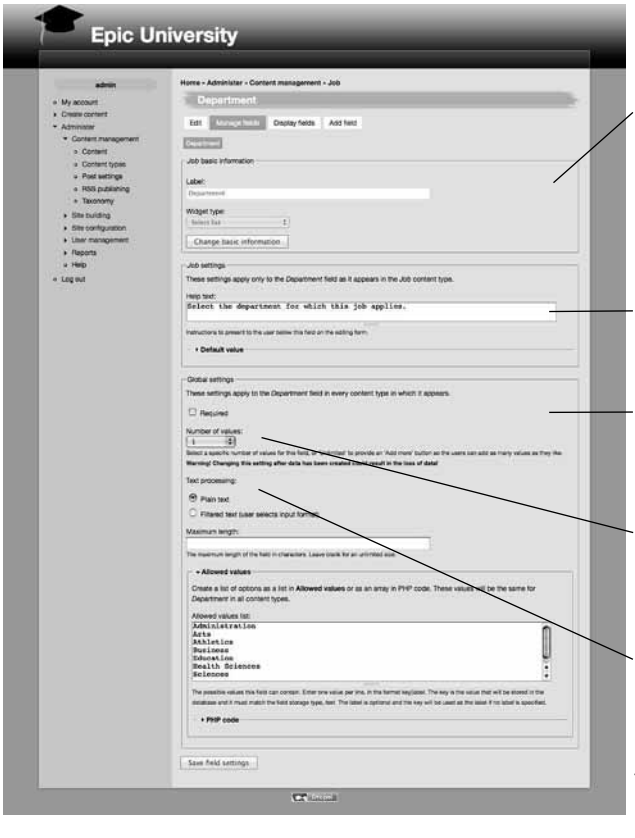
6. После сохранения нового поля щелчком на кнопке Save field settings (Сохранить настройки поля) вы должны вернуться на вкладку Manage fields (Управлять полями) (*admin/content/node-type/job/fields*). Теперь можно добавить поле Salary (Зарплата). Заполните форму добавления нового поля в соответствии с табл. 3.6. Мы определили поле зарплаты как целочисленное, но если вам потребуется указать центы, можно использовать тип поля Decimal.

Таблица 3.6. Параметры создания поля Salary (зарплата)

Поле	Значение
Label (Заголовок)	Salary

Field name (Имя поля)	salary
Select a field type (Выбор типа поля)	Integer
Select a widget (Выбор виджета)	Text field (Текстовое поле)

7. Щелкните на кнопке Save (Сохранить), чтобы создать поле Salary (Зарплата), и завершите настройку типа содержимого в соответствии с табл. 3.7, добавив описание и вывод префикса $ вместе с суммой зарплаты. По окончании щелкните на кнопке Save field settings (Сохранить настройки поля).



Показывает текст метки и тип виджета, выбранные в предыдущей форме.

При необходимости поля могут добавляться сразу в несколько типов содержимого. Данная группа настроек применяется к конкретному экземпляру поля только в данном типе содержимого.

Эти настройки применяются ко всем экземплярам поля, в каком бы типе содержимого оно ни находилось.

Настройки «Required» (Обязательно) и «Number of values» (Количество значений) присутствуют для всех полей и типов виджетов.

Остальные настройки могут зависеть от выбранного поля и типа виджета. В данном случае показаны настройки для поля типа Text, использующего виджет Select list. Для других полей здесь могут отображаться другие настройки.

Рис. 3.9. *Настройки поля CCK*

Таблица 3.7. Настройки поля Salary

Поле	Значение
Job settings (Настройки представления Job)	
Help text (Справочный текст)	Enter a yearly salary for this position. (Укажите сумму заработной платы за год для данной вакансии.)
Global settings (Общие настройки)	
Prefix (Префикс)	$

8. Добавьте поле Contact (Контактная информация) для этой вакансии. Обычно в этом поле будет указываться лицо, создавшее запись в каталоге вакансий, но мы даем пользователям возможность вводить в это поле сведения о любом сотруднике факультета, зарегистрированном на сайте. Это будет реализовано с помощью поля типа User reference (Ссылка на пользователя), которое будет отображаться как ссылка на другого пользователя сайта. Для большого сайта, где число сотрудников факультетов могло бы достигать нескольких сотен, для выбора можно вместо типа Select List (Раскрывающийся список) использовать тип Autocomplete Text Field (Текстовое поле с функцией автодополнения). Введите значения из табл. 3.8 в поля формы New field (Новое поле) и щелкните на кнопке Save (Сохранить).

Таблица 3.8. Параметры создания поля Contact

Поле	Значение
Label (Заголовок)	Contact
Field name (Имя поля)	contact
Select a field type (Выбор типа поля)	User reference (Ссылка на пользователя)
Select a widget (Выбор виджета)	Select list (Раскрывающийся список)

9. Настройте поле ссылки на пользователя так, чтобы оно могло ссылаться только на пользователей с ролью editor (которая присвоена сотрудникам факультета), как показано в табл. 3.9. Кроме того, мы оставили параметр Reverse link (Обратная ссылка) неотмеченным. Обратная ссылка будет отображаться в виде ссылки из профиля контактного пользователя на каждую вакансию, для которой он был определен как контактное лицо. Оставив этот параметр неотмечен-

ным, мы уменьшаем вероятность рассылки спама людьми, отправившими заявки сразу на несколько вакансий. Это сужает список потенциальных пользователей, которые могут быть выбраны. По завершении щелкните на кнопке Save field settings (Сохранить настройки поля).

Роль *editor* была заведена в профиле установки Job Website, а кроме того, был создан пользователь «editor» и несколько других учетных записей сотрудников факультета. Вы можете присвоить роль «editor» другим пользователям, перейдя на страницу Administer (Администрирование)→User management (Управление пользователями)→Roles (Роли) (*admin/user/roles*).

Таблица 3.9. Настройки поля Contact

Поле	Значение
Job settings (Настройки представления Job)	
Reverse Link (Обратная ссылка)	не отмечено
Help text (Текст справки)	Select the faculty member who is the primary contact responsible for hiring this position. (Выберите сотрудника факультета, назначенного ответственным за наем на работу по этой позиции.)
Global settings (Общие настройки)	
User roles that can be referenced (Роли пользователей, которые могут ссылаться)	editor
User status that can be referenced (Состояние пользователя, который может ссылаться)	Active (Активный)

10. И наконец, прежде чем пользователи смогут создавать элементы содержимого типа Job, их необходимо наделить правом создавать и редактировать новый тип содержимого. Добавьте разрешения для нового типа содержимого на странице Administer (Администрирование)→User management (Управление пользователями)→Permissions (Разрешения) (*admin/user/permissions*) в соответствии со значениями параметров в табл. 3.10 и щелкните на кнопке Save permissions (Сохранить права доступа).

Таблица 3.10. Настройка разрешений для типа содержимого Job

Разрешение: модуль node	**anonymous user (анонимный пользователь)**	**authenticated user (зарегистрированный пользователь)**	**editor (редактор)**	**site administrator (администратор сайта)**
create job content (создавать материал типа job)			отмечен	отмечен
delete any job content (удалять любой материал типа job)				отмечен
delete own job content (удалять собственный материал типа job)			отмечен	отмечен
edit any job content (изменять любой материал типа job)				отмечен
edit own job content (изменять собственный материал типа job)			отмечен	отмечен

Практика: настройка отображения поля

Для удобства часто требуется отображать формы и страницы содержимого в определенном порядке, с определенным форматированием, соответствующим отображаемым данным. Ниже рассказывается о некоторых незначительных настройках, которые определяют способ отображения полей.

Прежде чем приступать к тестированию нашей формы, необходимо изменить порядок следования полей, сделав его более логичным. Перейдите на страницу Administer (Администрирование)→Content management (Содержание)→Content types (Типы материалов) и щелкните на ссылке manage fields (управлять полями) для типа Job (*admin/content/node-type/job/fields*), после чего вы можете определить любой порядок следования полей, какой вам понравится. С помощью ярлыков перетаскивания, которые находятся слева в каждой строке, измените порядок следования строк в таблице, чтобы она выглядела, как показано на рис. 3.10 и затем щелкните на кнопке Save (Сохранить):

- Job title (Заголовок)
- Department (Факультет)
- Description (Описание)
- Salary (Зарплата)
- Contact (Контактная информация)
- Menu settings (Настройки меню)

Label	Name	Type	Operations
Job title			
Department	field_department	text	Configure Remove
Description			
Salary	field_salary	number_integer	Configure Remove
Contact	field_contact	userreference	Configure Remove
Menu settings (admin)			

Рис. 3.10. *Порядок следования полей для типа содержимого Job*

На этом создание типа содержимого Job практически закончено. Давайте посмотрим, как теперь выглядит наша форма. Зарегистрируйтесь на сайте как пользователь *editor* с паролем *oreilly* и создайте новое объявление о вакансии, перейдя на страницу Create content (Создать материал)→Job (*node/add/job*). Форма должна выглядеть, как показано на рис. 3.11 (если вы зайдете как пользователь *admin*, вы увидите дополнительные параметры, скрытые от других пользователей). Закончив заполнение формы, щелкните на кнопке Save (Сохранить).

Взгляните на содержимое после его создания; оно выглядит не очень привлекательно. На рис. 3.12 показано, как по умолчанию выглядит содержимое типа Job. Метки располагаются над значениями соответствующих полей, что делает страницу более длинной, чем она могла бы быть, а при выводе суммы заработной платы можно было бы для отделения групп разрядов использовать запятую.

К счастью, пакет CCK предоставляет несколько способов изменения отображения содержимого, самый простой из них – применение форматтеров CCK. Все модули в системе Drupal позволяют определять форматы отображения различных полей, и в том числе пакет CCK предоставляет несколько форматов, которых достаточно для большинства ситуаций. Выполните следующие шаги, чтобы сделать отображение содержимого типа Job немного более привлекательным:

1. Зарегистрируйтесь как пользователь *admin* с паролем *oreilly*, перейдите на страницу Administer (Администрирование)→Content management (Содержание)→Content types (Типы материалов) и щелкните на ссылке edit (изменить) для типа Job (*admin/content/nodetype/job*).

Щелчком на вкладке Display fields (Поля) перейдите к настройке параметров отображения полей в содержимом типа Job. Измените значения параметров в соответствии с табл. 3.11.

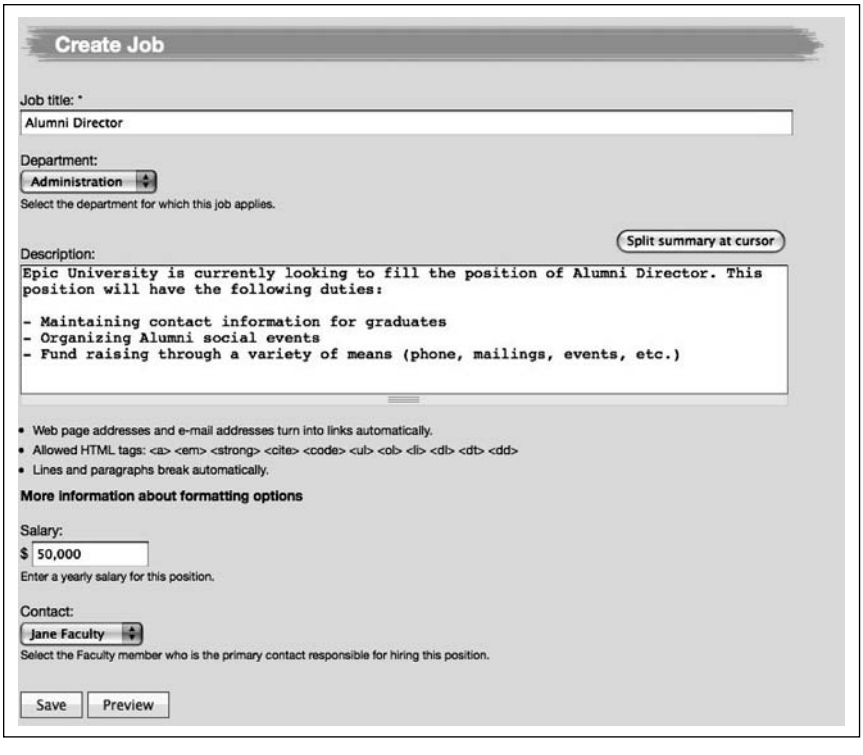


Рис. 3.11. *Форма создания вакансии, как ее видит пользователь, наделенный ролью editor*

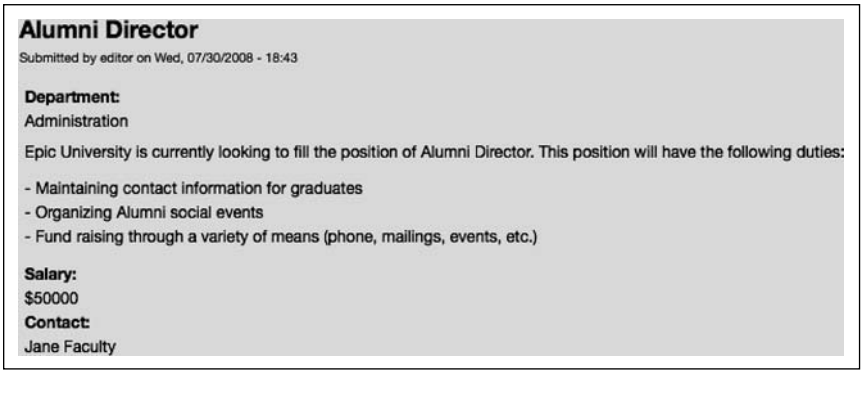


Рис. 3.12. *Вид содержимого типа Job по умолчанию*

Таблица 3.11. Настройки отображения полей для типа содержимого Job

Поле	Label (Заголовок)	Teaser (Анонс)	Full node (Весь материал)
Department	Inline (В одной строке)	Default (По умолчанию)	Default (По умолчанию)
Salary	Inline (В одной строке)	9,999	9,999
Contact	Inline (В одной строке)	<Hidden> (скрыто)	Default (По умолчанию)

2. После внесения изменений форма должна выглядеть, как показано на рис. 3.13.
3. Сохранив изменения, взгляните на элемент содержимого типа Job еще раз. Новый, более понятный вид показан на рис. 3.14.

Теперь можно видеть, что метки отображаются рядом со значениями, а не на отдельных строках, а в значения поля Salary добавлены запятые в качестве разделителя групп разрядов.

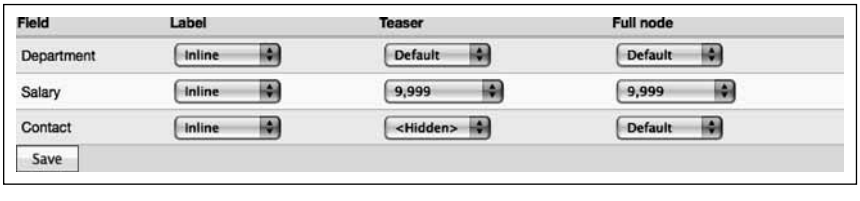


Рис. 3.13. *Настройки отображения полей для содержимого типа Job*

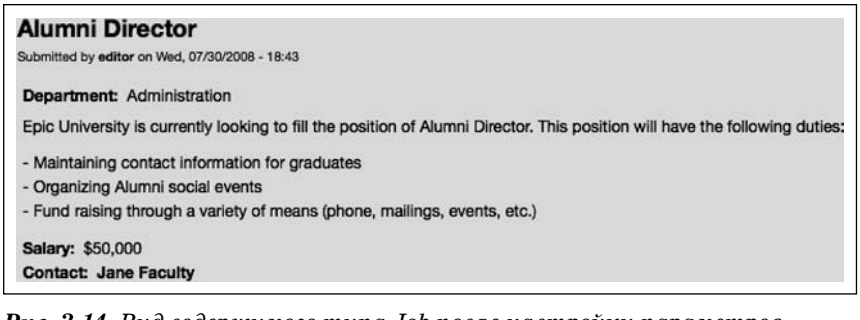


Рис. 3.14. *Вид содержимого типа Job после настройки параметров отображения*

Если вас не пугает (слишком сильно) возможность программирования на языке PHP и вы хотите получить более полный контроль над отображением типов содержимого, обратите внима-

ние на модуль Content Templates (Contemplate) (*http://drupal.org/project/contemplate*). Модуль Contemplate предоставляет альтернативный способ оформления (об оформлении будет рассказываться в главе 11) и обеспечивает веб-интерфейс редактирования программного кода, управляющего отображением типов содержимого, благодаря чему вы сможете определять, что будет появляться, как и где.

Практика: тип содержимого Job Application (заявка)

Теперь, когда появилась возможность публиковать на сайте университета объявления о вакансиях, было бы неплохо, если бы пользователи могли посылать свои резюме по заинтересовавшим их вакансиям. Мы создадим еще один тип содержимого, с именем Job Application или просто application (заявка) для краткости, определив в нем следующие поля:

- Title (Заголовок – обычное поле заголовка узла)
- Introductory message (Вводное сообщение – обычное поле содержания узла)
- Job (Поле ссылки на узел)
- Résumé (Поле файла)

1. Вернитесь на главную страницу настройки типов содержимого Administer (Администрирование)→Content management (Содержание)→Content types (Типы материалов) (*admin/content/types*). Создайте еще один тип содержимого, щелкнув на вкладке Add content type (Добавить тип содержимого) в верхней части страницы.
2. На странице Add content type (Добавить тип содержимого) заполните поля формы в соответствии с табл. 3.12. И опять мы без труда создадим первые два поля Title (Заголовок) и Introductory message (Вводный текст), использовав поля Title (Заголовок) и Body (Содержание), предоставляемые ядром Drupal.

Таблица 3.12. Параметры создания типа содержимого Job Application

Поле	Значение
Identification (Идентификация)	
Name (Имя)	Job Application
Type (Тип)	application
Description (Описание)	An application for a job position (Заявка на вакансию)

Submission form settings (Свойства формы)	
Title field label (Название поля заголовка)	Title (Заголовок)
Body field label (Название поля текста)	Introductory message (Вводный текст)

3. После щелчка на кнопке Save content type (Сохранить тип содержимого) и сохранения формы будет создан новый тип содержимого. Щелкните на ссылке manage fields (управлять полями) для типа Job Application (*admin/content/node-type/application/fields*). Используя значения из табл. 3.13, добавьте поле ссылки на узел типа Job. Это позволит связывать узлы типа Job Application с узлами типа Job.

Таблица 3.13. Параметры создания поля ссылки на узел типа Job

Поле	Значение
Label (Заголовок)	Job
Field name (Имя поля)	job
Select a field type (Выбор типа поля)	Node reference (Ссылка на узел)
Select a widget (Выбор виджета)	Select list (Раскрывающийся список)

4. Щелкните на кнопке Save (Сохранить) и настройте поле ссылки на узел типа Job в соответствии со значениями в табл. 3.14.

Таблица 3.14. Настройки для поля ссылки на узел типа Job

Поле	Значение
Global settings (Общие настройки)	
Required (Обязательно)	отмечено
Content types that can be referenced (Типы содержимого, на которые можно ссылаться)	Job

5. Последнее, что необходимо добавить в тип содержимого Job Application, – это дать пользователям возможность вместе с заявкой выгружать резюме или какие-либо другие файлы. Теоретически мы могли бы задействовать базовый модуль Upload, но вместо этого воспользуемся гибкими возможностями пакета CCK и задействуем поле типа file (файл), которое реализуется модулем FileField. Щелкните на ссылке manage fields (управлять полями) (*admin/content/node-type/*

application/fields), чтобы добавить и настроить поле типа file, используя значения из табл. 3.15, и затем щелкните на кнопке Continue (Продолжить).

Таблица 3.15. Параметры создания для Résumé (файл резюме)

Поле	Значение
Label (Заголовок)	Resume
Field name (Имя поля)	resume
Select a field type (Выбор типа поля)	File (файл)
Select a widget (Выбор виджета)	File Upload (поле выбора выгружаемого файла)

6. На следующей странице заполните поля с настройками в соответствии с табл. 3.16 и щелкните на кнопке Save field settings (Сохранить настройки поля). Нам необходимо ограничить перечень расширений файлов, допустимых для выгрузки на сервер, чтобы пользователи могли выгружать только файлы документов, а также указать, что все файлы, выгруженные с помощью виджета, должны сохраняться в подкаталоге *files*. Кроме того, модуль FileField позволяет управлять видимостью файлов в узле типа Job Application. В данном случае представленные настройки обеспечивают постоянную видимость файла, без возможности переопределить этот параметр. Однако возможные значения параметров, представленных здесь, учитывают случаи, когда такая гибкость необходима.

Таблица 3.16. Настройки для поля Résumé (файл резюме)

Поле	Значение
Job settings **(Настройки представления Job)**	
Permitted upload file extensions (Допустимые расширения выгружаемых файлов)	pdf doc txt rtf pages odf
Path settings > File path (Настройки пути > Путь к файлу)	Resumes
Global settings (Общие настройки)	
Required (Обязательно)	отмечено
Default list value (Перечисляется по умолчанию)	Listed (Перечисляется)

How should the list value be handled? (Как обрабатывается перечисление по умолчанию?)	Enforce Default (Принудительно использовать значение по умолчанию)
Description Field (Поле описания)	Disabled (Отключено)

Список поддерживаемых расширений файлов, используемый по умолчанию, при отображении автоматически выводится ниже поля выбора файла, поэтому нет необходимости дублировать эту информацию в тексте справки поля.

7. Теперь у нас имеются все необходимые поля. Измените порядок следования полей на вкладке Manage fields (Управлять полями), как показано ниже:
   - Title
   - Job
   - Introductory message
   - Résumé
   - Menu settings (Настройки меню)
8. В заключение добавим разрешения на доступ к новому типу содержимого, перейдя на страницу Administer (Администрирование)→User management (Управление пользователями)→Permissions (Разрешения) (*admin/user/permissions*). Нам необходимо, чтобы зарегистрированные пользователи могли управлять своими заявками, а редакторы могли управлять любыми заявками. Установите параметры настройки в соответствии со значениями в табл. 3.17 и щелкните на кнопке Save permissions (Сохранить права доступа).

Таблица 3.17. Настройка разрешений для типа содержимого Job Application

Разрешение: модуль node	**anonymous user (анонимный пользователь)**	**authenticated user (зарегистрированный пользователь)**	**editor (редактор)**	**site administrator (администратор сайта)**
create application content (создавать материал типа application)		отмечен		
delete any application content (удалять любой материал типа application)			отмечен	отмечен

Таблица 3.17 (продолжение)

Разрешение: модуль node	**anonymous user (анонимный пользователь)**	**authenticated user (зарегистрированный пользователь)**	**editor (редактор)**	**site administrator (администратор сайта)**
delete own application content (удалять собственный материал типа application)		отмечен		
edit any application content (изменять любой материал типа application)			отмечен	отмечен
edit own application content (изменять собственный материал типа application)		отмечен		

На этом мы закончили настройку формы ввода для содержимого типа Job Application. Теперь посмотрим, как выглядит эта форма для пользователя, обладающего ролью authenticated user (зарегистрированный пользователь). Зарегистрируйтесь как пользователь *user* с паролем *oreilly* и откройте страницу Create content (Создать материал)→Job Application (*node/add/job*), которая должна выглядеть, как показано на рис. 3.15.

Пользователь *user* был создан автоматически, после того как вы запустили профиль установки Job Posting.

Так как визуальное представление заявок на вакансии нам кажется менее важным, чем отображение информации о вакансиях, мы пропустим настройку параметров отображения для данного типа содержимого. Однако вы можете внести необходимые изменения на странице Administer (Администрирование)→Content management (Содержание)→Content types (Типы материалов)→Job Application→Fields (Поля) (*admin/content/node-type/application/fields*). После того как пользователь создаст новую заявку, она должна выглядеть, как показано на рис. 3.16.

Здесь важно заметить, как выглядит наше поле ссылки на узел (Job) после присвоения ему значения. По умолчанию оно будет выглядеть как ссылка на исходный элемент содержимого. Щелчок на ссылке Alumni

Director перенесет нас на страницу с объявлением о вакансии Alumni Director. Существуют и другие способы отображения полей ссылок на узлы, которые можно увидеть на вкладке Display fields (Поля), на странице с настройками типа содержимого Job Application (Заявка) (*admin/content/node-type/application/display*).

Сейчас было бы неплохо наполнить сайт некоторым содержимым. Зарегистрируйтесь как пользователь *admin* или *editor* с паролем *oreilly* и создайте несколько элементов содержимого типа Job (объявлений о вакансиях) на странице Create content (Создать материал)→Job (*node/add/job*). Также неплохо было бы создать несколько заявок на разные вакансии. Наличие этого содержимого пригодится вам в следующем разделе.

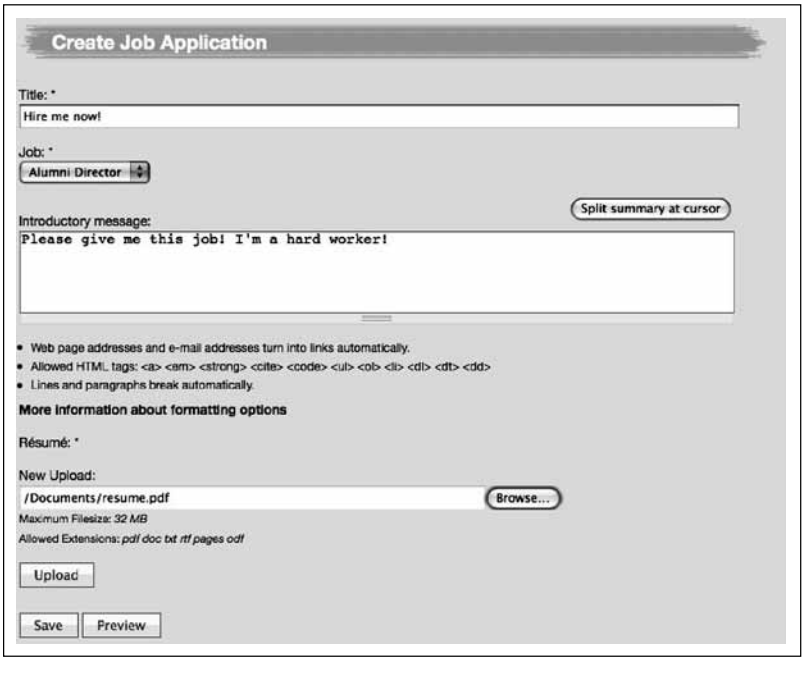


Рис. 3.15. *Форма ввода заявки на вакансию, как ее видит любой зарегистрированный пользователь*

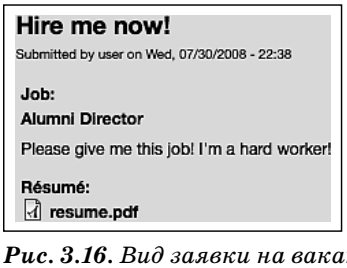


Рис. 3.16. *Вид заявки на вакансию*

В центре внимания: модуль Views

Модуль Views позволяет получать списки данных, находящихся на вашем сайте: пользователей, комментариев, узлов и много другого. Любой список, созданный модулем Views, называется *представлением*. На рис. 3.17 показаны примеры списков, которые можно получить с помощью модуля Views.

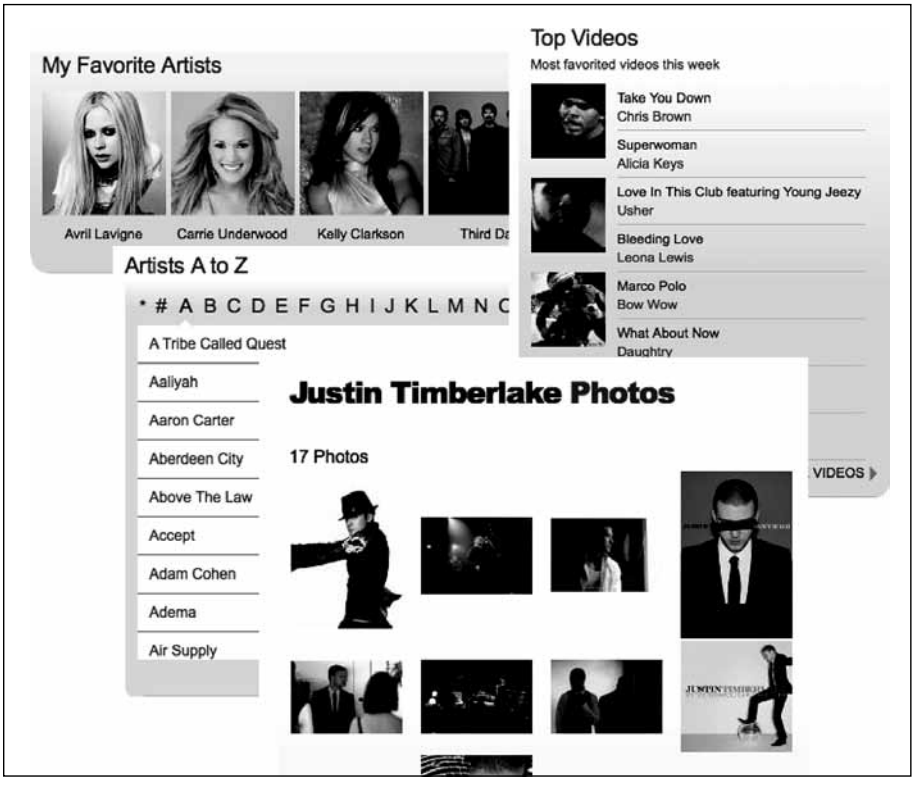


Рис. 3.17. *Примеры представлений, созданных с помощью модуля Views*

Чтобы создать простое представление, необходимо определить: какие поля отображать (заголовок узла, имя автора, изображения и прочие), как должен фильтроваться этот список (например, отображать только опубликованные узлы типа Story (Статья)), порядок сортировки списка (например, в обратном хронологическом порядке) и как должен выглядеть список при отображении (например, как блок маркированного списка заголовков).

С точки зрения технической реализации, модуль Views – это визуальный построитель запросов SQL. Когда вы создаете представление, вы фактически конструируете запрос, с помощью которого модуль Views будет извлекать информацию из базы данных сайта. У модуля Views

имеются существенные преимущества перед запросами, встроенными непосредственно в программный код. Например:

- Для создания списка содержимого не придется писать какой-либо код.
- Модули сами сообщают модулю Views о своих полях, поэтому вам не обязательно знать структуру базы данных и вы можете не волноваться о проблемах, связанных с изменением этой структуры при обновлении модулей.
- Одно и то же представление может использоваться в нескольких местах на сайте, как в блоках, так и на страницах.
- Результаты можно представлять в виде многостраничных списков, предусматривать возможность сортировки по столбцам, использовать средства AJAX для постраничного вывода или фильтрации, чтобы позволить посетителям исследовать содержимое с желательной степенью детализации.

Язык запросов SQL и модуль Views

SQL – это язык запросов к базе данных, позволяющий извлекать информацию из базы данных. Язык SQL состоит из простых команд, таких как:

```
SELECT title FROM node WHERE nid = 10
```

Каждая из этих команд называется *запросом*. Запросы на извлечение необходимой информации из базы данных могут быть намного более длинными, и это одна из причин привлекательности модуля Views: он может создавать запросы за вас.

Представление основано на использовании запроса SQL, поэтому многие понятия в модуле Views отображаются непосредственно на SQL. Рассмотрим основные части запроса: инструкция **select**, предложение **where** и предложение **order by**. Они непосредственно соответствуют полям, фильтрам, критериям сортировки и другим компонентам представлений, рассматриваемым ниже в этой главе.

```
SELECT [поля]
FROM [тип представления и любые отношения]
WHERE [фильтры или аргументы]
ORDER BY [критерии сортировки]
```

Чтобы использовать модуль Views, вам не требуется знать язык SQL, тем не менее, они настолько тесно связаны между собой, что знакомство с языком SQL может существенно помочь в понимании модуля Views или упростить преобразование существующего программного кода в представления.

Более, чем что бы то ни было, модуль Views может существенно ускорить разработку сайта, не требуя при этом обладания знаниями о разработке модулей или коротких сценариев на языке PHP. Модуль Views может образовать широкую магистраль вывода содержимого вашего сайта.

В отличие от создания нового типа содержимого, весь процесс создания представления проходит на одной странице. На рис. 3.18 изображена основная часть интерфейса, в котором производится создание представления.

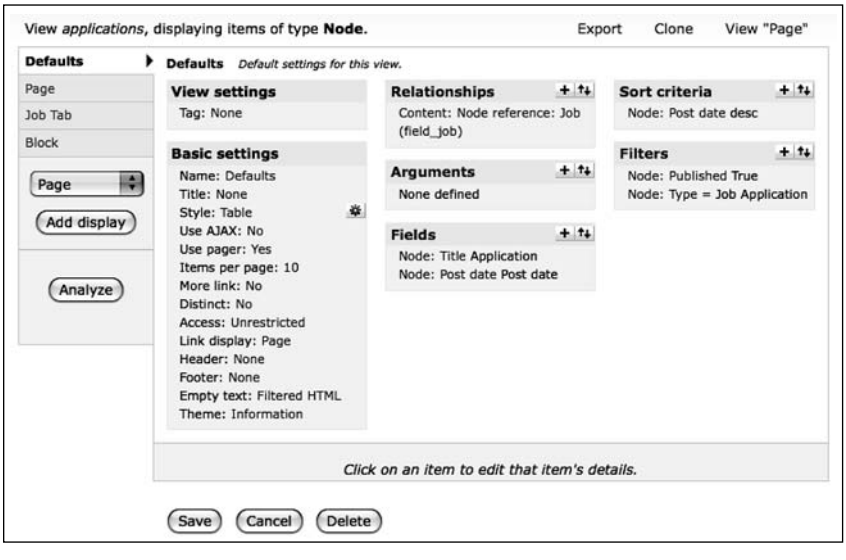


Рис. 3.18. *Интерфейс создания представления*

Отображения

Отображение (*display*) определяет, как будет отображаться представление. Представление может обладать несколькими отображениями и даже создавать несколько страниц, где различными способами перечисляется одна и та же информация. Крайняя левая колонка в интерфейсе к модулю Views дает возможность выбрать отображение. На рис. 3.19 показано добавление в представление нового отображения.

По умолчанию имеется четыре типа отображений, каждый из которых преследует определенную цель, при этом другие модули могут добавлять дополнительные отображения:

Attachment (прикрепленный файл)

Вспомогательное отображение, которое может прикрепляться поверх и/или ниже других типов отображений. Удобно использовать для организации контекста представления или добавления глоссария, когда предусматривается фильтрация представления.

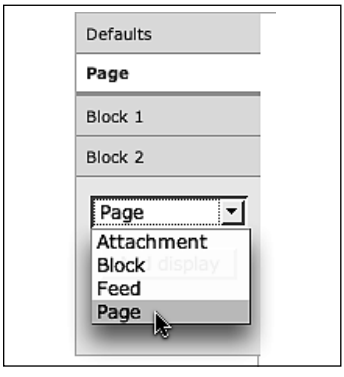


Рис. 3.19. *Добавление нового отображения*

Block (блок)

Создает компактное отображение списка, которое можно поместить в боковую колонку или в любую другую область, на странице настройки блоков Administer (Администрирование)→Site building (Конструкция сайта)→Blocks (Блоки) (*admin/build/block*).

Feed (рассылка RSS)

Создает настраиваемую рассылку RSS, на которую можно подписаться с помощью программ чтения RSS. Рассылки могут иметь собственный адрес URL и присоединяться к любому отображению типа Block или Page.

Page (страница)

Создает страницу с собственным адресом URL, где представление занимает область основного содержимого.

Представления имеют множество интересных параметров настройки, облегчающих настройку отображения содержимого. Параметры для всех типов отображения могут настраиваться все сразу на общей вкладке Defaults (По умолчанию) или для каждого типа отображения в отдельности, позволяя переопределить настройки представления по умолчанию. Чтобы изменить какое-либо значение в интерфейсе модуля Views, щелкните на ссылке, представляющей параметр; в результате ниже появится форма настройки этого параметра. Помимо изменения отдельных значений некоторые параметры могут потребовать дополнительной настройки. В этом случае рядом с параметром отображается маленькое изображение шестеренки, щелчок на которой отобразит форму с дополнительными настройками. Понятия значений по умолчанию и параметров настройки иллюстрируются на рис. 3.20.

Обращайте особое внимание при настройке определенного типа отображения, так как между разными типами отображений настройки могут несколько отличаться. Некоторые из наиболее существенных

параметров настройки представления доступны только при конфигурировании определенного типа отображения. Например, чтобы определить адрес URL для представления, необходимо настроить отображение Page (Страница). Адрес URL представлен отдельным параметром внутри группы настроек Page settings (Параметры страницы), которая отображается в нижней части слева. Параметры настройки, доступные для всех типов отображений, показаны на рис. 3.21.

Особенно *пристальное* внимание обращайте на шрифт – курсивный шрифт обозначает, что данное значение используется по умолчанию. Вы обязательно должны переопределять значения (щелчком на кнопке Override (Переопределить)), предназначенные для конкретного отображения, в противном случае будут использоваться значения по умолчанию независимо от того, какое отображение в настоящий момент выбрано.

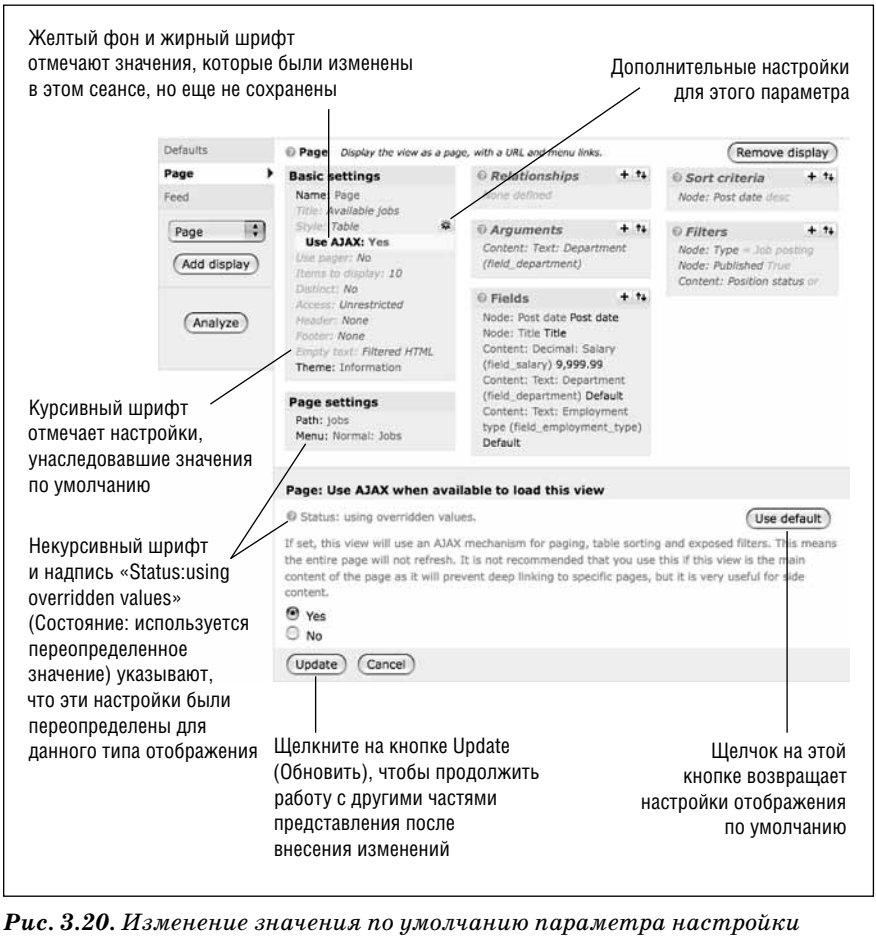


Рис. 3.20. *Изменение значения по умолчанию параметра настройки*

Рис. 3.21. *Перечень доступных параметров может изменяться в зависимости от типа отображения*

Элементы представления

Существенной частью настройки представления является определение содержимого, которое требуется отобразить. Эта настройка включает в себя поля, фильтры, аргументы, критерии сортировки и отношения. Каждый из этих элементов будет подробно рассматриваться в следующем разделе. Эти настройки находятся в интерфейсе модуля Views справа, как показано на рис. 3.22.

Если группа параметров Fields (Поля) не видна, это может быть обусловлено необходимостью установить параметр отображения Row style (Стиль строки) в значение Fields (Поля) вместо Node (Материал).

Поля

Поле представляет элемент данных, который требуется отобразить. Примерами полей могут служить заголовок узла, адрес электронной почты пользователя, значение поля CCK, рубрика таксономии или любые другие данные в системе Drupal. Альтернативой выбора для отображения отдельных полей (Fields) является выбор значения Node (Материал) в параметре Row style (Стиль строки), что позволяет отображать выбранные узлы в виде списка, получив отображение, похожее на главную страницу в системе Drupal.

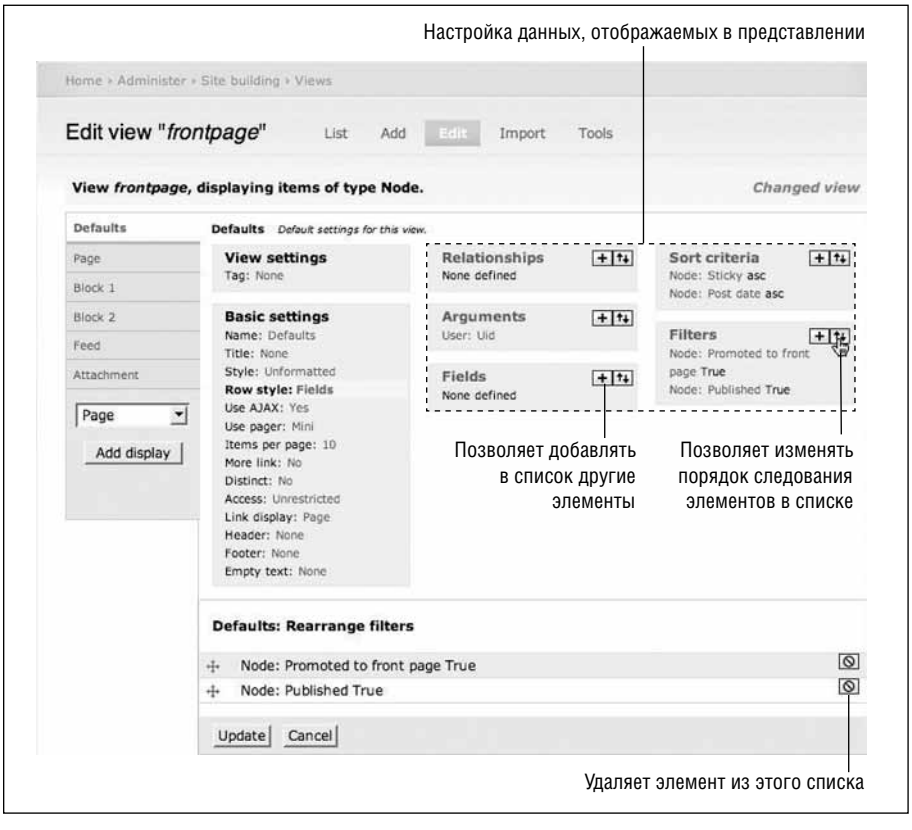


Рис. 3.22. *Настройка элементов содержимого, включаемых в представление*

Фильтры

По умолчанию модуль Views отображает все комментарии и узлы, имеющиеся на веб-сайте, доступные пользователям. Чтобы ограничить отображаемое содержимое с помощью каких-либо критериев, используются фильтры. Некоторые наиболее типичные фильтры позволяют просматривать только те узлы, для которых включен флаг Published (Опубликовано), или только узлы определенного типа, такие как наши узлы Job (Вакансия) или Job Application (Заявка).

Аргументы

Аргументы – это динамическая версия фильтров. В случае когда требуется создать список материалов, относящихся к определенному пользователю, было бы нежелательно создавать отдельное представление для каждого пользователя сайта. Вместо этого модуль Views позволяет создать единственное представление и фильтровать результаты по идентификационному номеру пользователя, передавая его как аргумент.

Конструирование эффективных представлений

Представление, использующее значение Node (Материал) в параметре Row style (Стиль строки), обычно менее эффективно, чем представление, использующее значение Field (Поле). Это обусловлено тем, что при работе с полями модуль Views способен обращаться к данным напрямую, тогда как при выводе списка узлов приходится загружать все поля для каждого узла, который требуется отобразить. Например, представление, которое отображает только поля заголовка и автора узла, при использовании значения Fields (Поля) позволяет избежать ненужной загрузки рубрик таксономии, полей CCK и любых других данных, добавляемых дополнительными модулями.

Даже загрузка большого количества полей при использовании значения Fields (Поля) в настройках отображения часто оказывается эффективнее, потому что модуль Views может извлекать все необходимые данные за один запрос, тогда как при загрузке узлов по отдельности обычно выполняется не менее 10 запросов на узел, в зависимости от количества включенных модулей.

Модуль Views выводит некоторую полезную для разработчиков информацию в нижней части интерфейса, включая сгенерированный запрос SQL и время, которое требуется, чтобы сгенерировать представление. Кнопка Analyze (Анализ) поможет вам обратить внимание на вещи, которые вы могли упустить из виду. Вы можете использовать эту информацию, чтобы видеть, как влияют на производительность вносимые изменения.

Как правило, аргументы передаются в строке URL. Если доступ к представлению производится по адресу URL *http://example.com/my_view*, то элементы пути, следующие за *my_view*, могут интерпретироваться как аргументы. В строке URL *http://example.com/my_view/10* число *10* может быть первым аргументом. Вы можете определить в своем представлении столько аргументов, сколько потребуется.

В дополнение к аргументам в конце строки URL имеется возможность помещать аргументы в середине URL, если использовать символ % в настройках аргументов. Такая возможность может быть полезной в случае необходимости использовать некоторые из существующих путей в Drupal, такие как путь к пользователю, который может выглядеть так: *http://example.com/user/10/my_view*. Нам по-прежнему требуется, чтобы число 10 было первым аргументом, но чтобы на этот раз аргумент был выбран из середины URL. Если определить путь URL к представлению, как *user/%/my_view*, символ % будет замещен содержимым URL и передан представлению как первый аргумент. Если сей-

час вам не все понятно, не волнуйтесь, – мы рассмотрим пример использования аргументов такого типа в следующем разделе «Практика: модуль Views».

Критерии сортировки

Сузив круг результатов, извлекаемых из базы данных, и получив значения полей, которые требуется отобразить, вы можете задействовать критерии сортировки, чтобы определить порядок отображения результатов. Например, можно отсортировать материалы по дате создания, по имени пользователя автора или по рубрикам таксономии.

Отношения

Отношения впервые появились в версии модуля Views для Drupal 6. Когда необходимо включить в представление данные из другого объекта, недоступные непосредственно (например, информация о пользователе) внутри списка содержимого (основанного на узлах), отношения позволят вам извлекать информацию из объектов, которые связаны с содержимым в списке. Отношения в представлении можно рассматривать как эквивалент предложений `JOIN` языка SQL, используемого для определения запросов к реляционным базам данных.

Мы представим пример использования отношения, в котором заявка на вакансию будет связана с определенными объявлениями о вакансиях. Пользователь будет создавать элемент содержимого (заявку), связанный с другим элементом содержимого (объявлением). Используя отношения модуля Views, мы сможем создавать списки материалов, которые включают информацию как из заявки, так и из объявления.

Практика: модуль Views

Наш сайт требует создания двух представлений. Одно представление должно отображать все имеющиеся вакансии для пользователей сайта. Сотрудники факультетов (если говорить более точно – пользователи, наделенные ролью «editor») будут пользоваться вторым представлением для просмотра списка заявок на различные вакансии.

Для использования модуля Views необходимо включить его. Модуль Views состоит из двух частей: сам модуль Views, выполняющий основную, «низкоуровневую», работу, и модуль Views UI, который отвечает за формирование страниц интерфейса, используемых для настройки представлений. Кроме того, имеется дополнительный модуль Advanced Help, обеспечивающий вывод встроенных справочных сообщений для таких модулей, как Views.

1. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Modules (Модули) (*admin/build/modules*) и включите следующие модули:

- Other (Другой)
  - Advanced Help
- Views
  - Views
  - Views UI

2. Настройте разрешения для модуля на странице Administer (Администрирование)→User management (Управление пользователями)→Permissions (Разрешения) (*admin/user/permissions*), используя значения из табл. 3.18. Только администраторы будут иметь возможность настраивать модуль Views, поэтому доступ к особенностям модуля Advanced Help требуется разрешить только им.

Таблица 3.18. Разрешения для модуля Views

Разрешение	**anonymous user (анонимный пользователь)**	**authenticated user (зарегистрированный пользователь)**	**editor (редактор)**	**site administrator (администратор сайта)**
модуль advanced_help				
view advanced help index (просматривать предметный указатель справки)				отмечено
view advanced help pop-up (просматривать диалоги справки)				отмечено
view advanced help topic (просматривать темы справки)				отмечено
модуль views				
administer views (управлять представлениями)				отмечено
модуль views_ui				
access all views (иметь доступ ко всем представлениям)				отмечено

Будьте внимательны при обращении с разрешением access all views (иметь доступ ко всем представлениям) – пользователи, обладающие этим разрешением, смогут обходить все ограничения, которые вы наложите на представления, то есть они смогут просматривать все представления.

Представление Jobs (вакансия)

Представление «jobs» отображает список вакансий университета Epic University по факультетам. Внешний вид представления изображен на рис. 3.23.

1. Для начала перейдите на страницу настройки модуля Views: Administer (Администрирование)→Site building (Конструкция сайта)→Views (*admin/build/views*).
2. Щелкните на вкладке Add (Добавить) вверху страницы и заполните форму значениями из табл. 3.19. Это самая обычная информация, которую потребуется указывать для любого представления: название, описание и тип. Параметр tag (рубрика) является необязательным – он помогает фильтровать список представлений на главной странице, что может быть удобно для быстрого отображения списка определенных представлений. Заполненная форма должна выглядеть, как показано на рис. 3.24.

Таблица 3.19. Настройки, используемые при создании нового представления jobs

Параметры настройки представления	**Значение**
View name (Название представления)	jobs
View description (Описание представления)	A list of available positions at Epic University (Список доступных вакансий университета Epic University)
View tag (Рубрика представления)	jobs
View type (Тип представления)	Node (Материал)

3. После щелчка на кнопке Next (Далее) перед вами откроется основная страница конструирования представления, где можно определить, как будет отображаться представление по умолчанию. Установите настройки в разделе Basic Settings (Базовые настройки) в соответствии со значениями в табл. 3.20 и как показано на рис. 3.25. После установки значения каждого параметра необходимо щелкнуть на кнопке Update (Обновить), чтобы сохранить его.

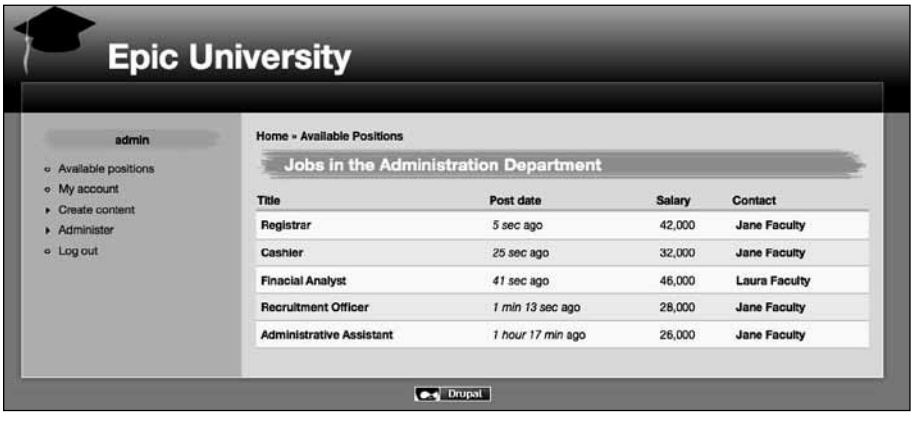


Рис. 3.23. *Примерный вид страницы из представления jobs со списком вакансий факультета Administration*

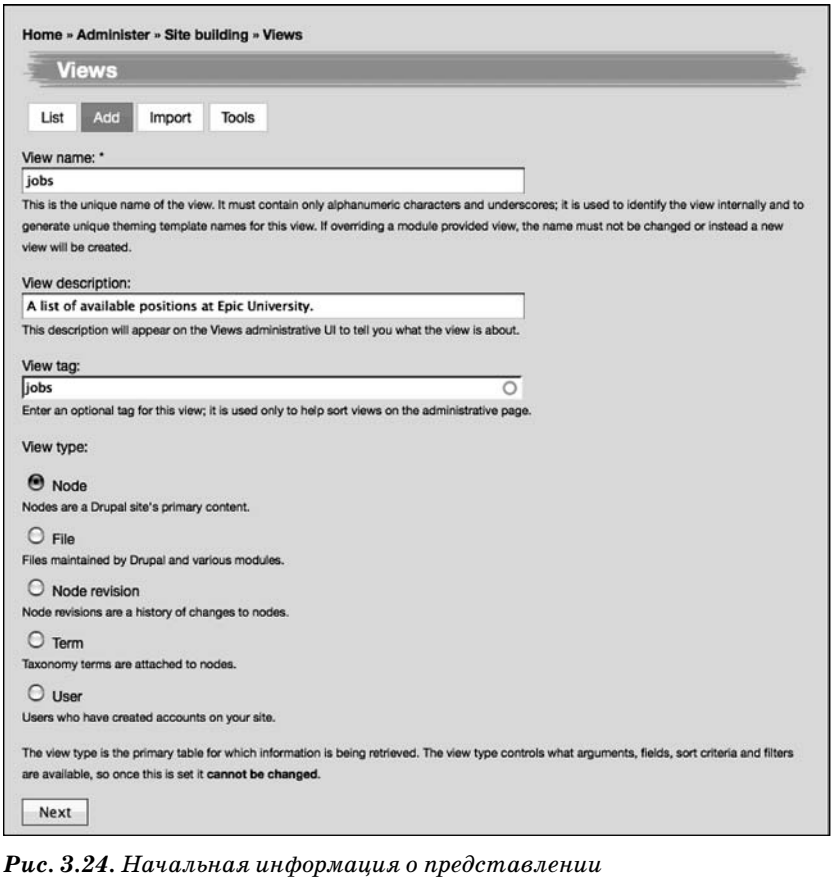


Рис. 3.24. *Начальная информация о представлении*

Рис. 3.25. *Базовые настройки представления Available Positions*

Указанные настройки определяют представление, которое отображает информацию в виде таблицы, снабженной элементами управления, позволяющими переходить с одной страницы на другую. В случае отсутствия вакансий представление выводит указанное в настройках сообщение.

Таблица 3.20. Базовые настройки представления jobs

Defaults: Basic settings (базовые настройки)	**Значение**
Title (Название)	Available Positions (Доступные вакансии)
Style (Стиль)	Table (Таблица)
Use pager (Использовать компонент выбора страниц)	Yes (Да) Full pager (Полнофункциональный компонент)
Empty text (Текст для пустого представления)	There are currently no positions available. (В настоящее время нет ни одной свободной вакансии.)

4. В этот момент вы можете получить сообщение об ошибке: «Display Defaults uses fields but there are none defined for it or all are excluded» («Отображение по умолчанию использует поля, но пока ни одно поле не определено либо все поля исключены»). Исправим эту ошибку, добавив в представление несколько полей. Для вакансии нам потребуется отобразить заголовок, дату публикации объявления, размер зарплаты и контактную информацию.

   Чтобы приступить к добавлению новых полей, щелкните на ярлыке с изображением знака плюс в области Fields (Поля). Добавьте поля в соответствии с табл. 3.21, как показано на рис. 3.26. Для ускорения ввода можно выбрать значения Node (Материал) и Content (Содержание) в раскрывающемся списке Groups (Группы), чтобы оставить

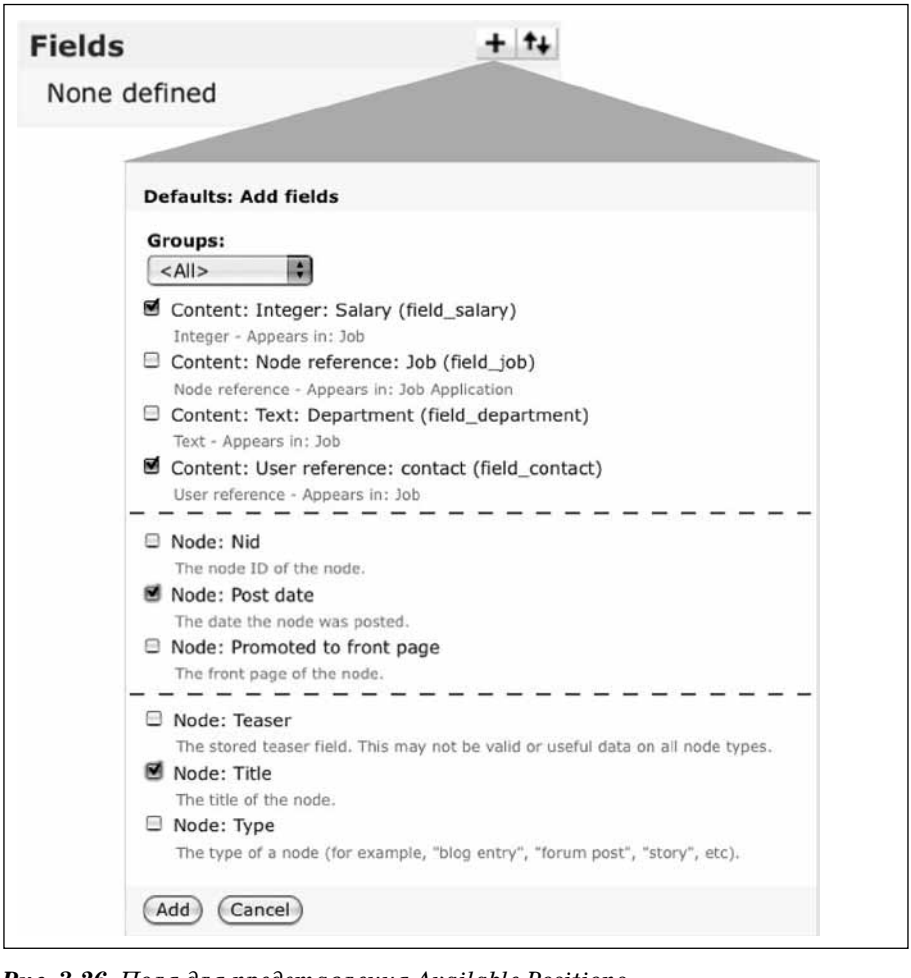


Рис. 3.26. *Поля для представления Available Positions*

в списке доступных полей только те, которые перечислены в этой группе.

Таблица 3.21. Базовые настройки полей для представления jobs

Defaults: Add fields (по умолчанию: добавить поля)	**Значение**
Node: Title (Узел: Заголовок)	отмечено
Node: Post date (Узел: Дата создания)	отмечено
Content: Integer: Salary (Содержание: Integer: Salary)	отмечено
Content: User reference: Contact (Содержание: Ссылка на пользователя: Contact)	отмечено

Поля в группе Node (Материал) представляют свойства, общие для всех типов узлов, такие как автор узла или дата создания. Поля в группе Content (Содержание) являются дополнительными и добавляются пакетом CCK.

5. После щелчка на кнопке Add (Добавить) модуль Views отобразит по очереди формы для настройки полей, что позволит настроить параметры каждого поля. Закончив ввод значений из табл. 3.22, как показано на рис. 3.27, щелкните на кнопке Update (Обновить).

Таблица 3.22. Настройки отдельных полей для представления jobs

Defaults: Configure field setting (по умолчанию: параметры настройки полей)	**Значение**
Content: Integer: Salary (Содержание: Integer: Salary)	Format (формат): 9,999
Content: User reference: Contact (Содержание: Ссылка на пользователя: Contact)	(оставить значение по умолчанию)
Node: Post date (Узел: Дата создания)	Date format (Формат даты): Time ago (Прошло времени)
Node: Title (Узел: Заголовок)	Link this field to its node (Связать это поле с указанным узлом): отмечено

6. Щелкните на ярлыке, обозначающем операцию переупорядочения, в разделе Fields (Поля) и расставьте поля в следующем порядке:

- Node: Title
- Node: Post date
- Content: Integer: Salary
- Content: User reference: Contact

7. Теперь, когда в представление были добавлены поля, необходимо сохранить его, щелкнув на кнопке Save (Сохранить). Если эта кнопка окрашена в серые тона, возможно, вы не завершили операцию редактирования какого-либо поля; завершите ее, после чего кнопка станет доступна.

Рис. 3.27. *Настройки полей для представления Available Positions*

8. Теперь наше представление отображает все необходимые данные, но в нем пока нет никаких ограничений на то, какие типы содержимого отображать, а кроме того, объявления отображаются в случайном порядке. Прокрутив страницу интерфейса настройки представления, вы увидите область оперативного предварительного просмотра представления, настраиваемого в настоящий момент, – она должна выглядеть, как показано на рис. 3.28.

   Как видно, представление пока никак не фильтруется и включает в себя не только объявления о вакансиях. В список включено наше

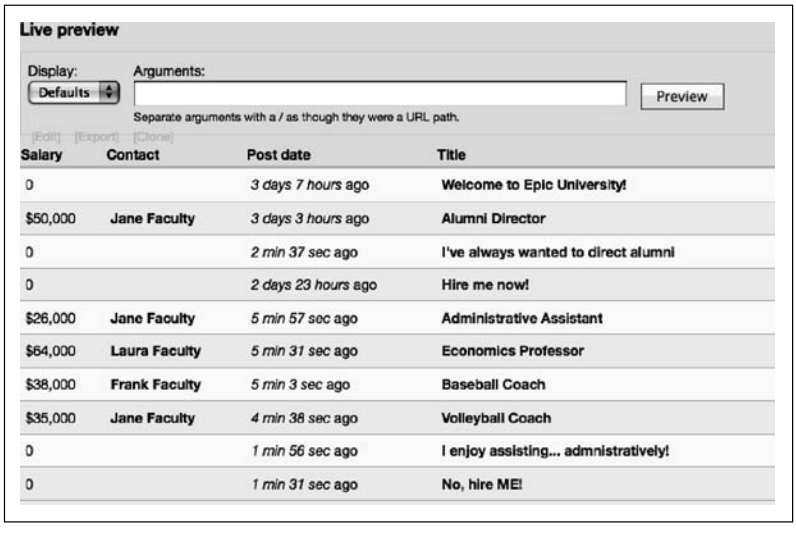


Рис. 3.28. *Наше представление, пока еще без поддержки фильтрования*

приветствие (которое является частью содержимого страницы), а также все заявки на вакансии, оставленные на сайте.

Это не проблема! Чтобы добавить в представление несколько новых фильтров, щелкните на ярлыке с изображением знака плюс в области Filters (Фильтры). Как и в случае с полями, здесь также можно использовать раскрывающийся список Groups (Группы), чтобы отфильтровать список и оставить в нем только требуемые элементы. После установки требуемых флажков и щелчка на кнопке Add (Добавить) под каждым элементом будет отображаться форма с настройками, которые можно циклически обойти, щелкая на кнопке Update (Обновить). Добавьте фильтры и настройки в соответствии с табл. 3.23 и как изображено на рис. 3.29.

Таблица 3.23. Фильтры по умолчанию для представления jobs

Defaults: Add filters (По умолчанию: Добавить фильтры)	**Значение**
Node: Published (Узел: Опубликовано)	Published (Опубликовано): отмечено
Node: Type (Узел: Тип)	Operator (Оператор): Is one of (Один из) Node type (Тип узла): Job

9. Вернувшись к оперативному предварительному просмотру, можно убедиться, что теперь нежелательные элементы исключены из представления.

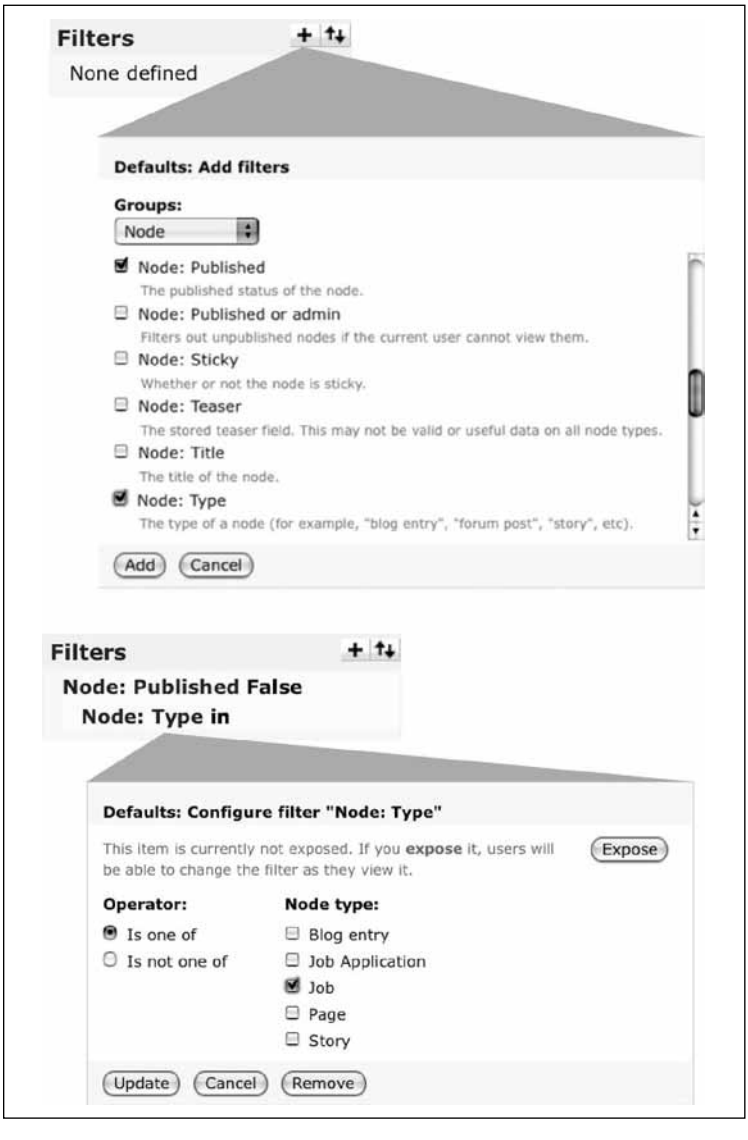


Рис. 3.29. *Настройка фильтра для представления Available Positions*

Если конструируемое представление не предназначено для нужд администрирования и просмотра неопубликованного содержимого, вам почти всегда будет требоваться использовать фильтр Node: Published (Материал: Опубликовано), чтобы представление всегда отображало только то содержимое, которое предназначено для просмотра посетителями сайта. Щелчок на кнопке Analyze (Анализ) по окончании создания представления позволит вам заранее заметить эту и другие типичные проблемы.

10. Дополнительно нам необходимо настроить критерий сортировки, чтобы разместить эти элементы в логическом порядке. Щелкните на ярлыке с изображением знака плюс в области Sort criteria (Критерий сортировки) и настройте параметры сортировки в соответствии

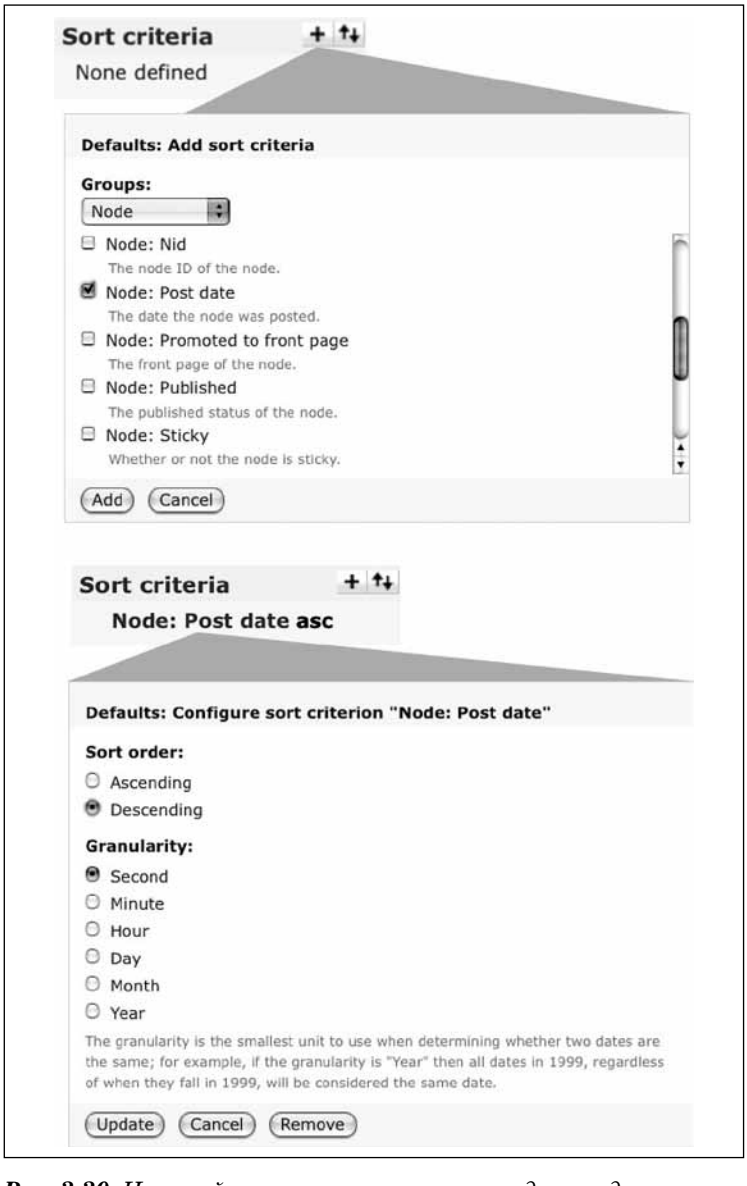


Рис. 3.30. *Настройка критерия сортировки для представления Available Positions*

с табл. 3.24 и рис. 3.30, а затем щелкните на кнопке Update (Обновить), чтобы сохранить настройки.

Таблица 3.24. Критерий сортировки по умолчанию для представления jobs

Defaults: Sort criteria (критерий сортировки)	**Значение**
Node: Post date (Узел: Дата создания)	Sort order: Descending (Порядок сортировки: По убыванию) Granularity: Second (Точность: Секунда)

Настройки Granularity (Точность) обеспечивают удобную возможность одинаковой сортировки всех элементов, созданных в течение одного и того же периода времени. Эта возможность, например, позволит вам сортировать новые объявления о вакансиях сначала по дню публикации, а затем, в пределах каждого отдельного дня, отсортировать заголовки по алфавиту. Мы оставили значение Second (секунда) в этом параметре просто потому, что это наиболее эффективный метод, так как в базе данных даты создания объявлений хранятся в секундах.

11. Наше представление почти готово. Оно отображает необходимую информацию в виде таблицы, сортирует объявления по дате создания. Однако, чтобы фактически сделать это представление видимым для любых пользователей, необходимо добавить новое отображение. В левой части интерфейса выберите значение Page (Страница) в раскрывающемся списке и щелкните на кнопке Add display (Добавить отображение).
12. После добавления нового отображения в области Basic settings (Базовые настройки) появятся настройки этого отображения. Любые настройки, которые мы определим здесь, будут добавлены или изменят настройки по умолчанию, которые мы уже создали.

    В области Page settings (Настройки страницы) определите значения параметров настройки в соответствии с табл. 3.25, как показано на рис. 3.31.

Таблица 3.25. Настройки страницы для представления jobs

Defaults: Page settings (Настройки страницы)	**Значение**
Path (Путь)	jobs
Menu (Меню)	Type (тип): Normal menu entry (Обычный элемент меню) Title (заголовок): Available positions

Если вы не видите заголовок Page settings (Настройки страницы) группы параметров, вероятно, вы все еще просматриваете настройки Defaults display (Настройки отображения по умолчанию). В этом случае просто щелкните на ссылке Page (Страница).

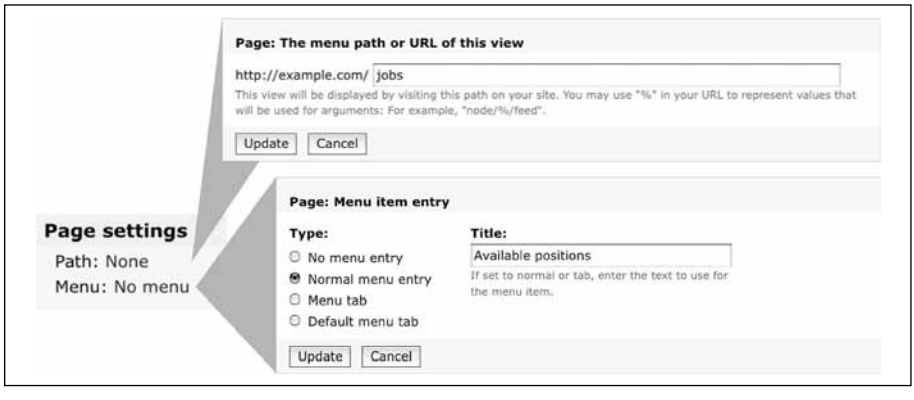


Рис. 3.31. *Настройки страницы для представления Available Positions*

13. После щелчка на кнопке Save (Сохранить) для модуля Views в навигационном меню должна появиться ссылка на представление Available Positions (Доступные вакансии). Эта ссылка должна выглядеть, как показано на рис. 3.32.

Рис. 3.32. *Новое представление с вакансиями и ссылка в навигационном меню*

14. Вы должны были заметить, что мы пока еще не включили в этот список поле Department (Факультет), это обусловлено тем, что в наших требованиях указано, что список должен иметь вид каталога. То есть на первой странице представления вакансий должен отображаться список факультетов, а затем, после щелчка на названии факультета, должен выводиться список всех имеющихся вакансий для этого факультета. Было бы весьма утомительно создавать отдельные представления для каждого факультета, поэтому мы реализуем это требование с помощью *аргумента* – мощной особенности модуля Views.

Прежде чем приступить к реализации такого списка, вернитесь назад, на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Views (*admin/build/views*) и щелкните на ссылке edit (изменить) для представления jobs. Щелкните на ярлыке с изображением знака плюс в области Arguments (Аргументы), чтобы добавить новый аргумент в отображение Page (Страница). Как и прежде, щелкните на кнопке Add (Добавить), чтобы добавить аргумент Department, и введите настройки в соответствии с табл. 3.26, как показано на рис. 3.33.

Таблица 3.26. Аргументы настройки страницы для представления jobs

Page: Arguments (Страница: Аргументы)	**Значение**
Content: Text: Department (Содержание: Текст: Department)	Title: Jobs in the %1 Department (Заголовок: Вакансии в отделе %1) Action to take if argument is not present: Summary, sorted ascending (Действие, выполняемое в случае отсутствия аргумента: Сводный список, отсортированный по возрастанию) Case: Capitalize each word (Регистр: каждое слово начинать с заглавного символа) Case in path: Lowercase (Регистр элементов пути: Нижний) Transform spaces to dashes in URL (Преобразовывать пробелы в строке URL в дефисы): отмечено Summary style (after clicking Update) (Стиль списка (после щелчка на кнопке Обновить)): List (Список)

Символы «%1» в заголовке выглядят немного непонятно – они будут замещены названием факультета динамически, при просмотре страницы. Параметр Action to take if argument not present (Действие, выполняемое в случае отсутствия аргумента) определяет, что должно делать представление, когда выполняется обращение по адресу URL *http://www.example.com/jobs*, а не по адресу, например, *http://www.example.com/jobs/athletics*. Здесь мы предписываем отображать список факультетов, отсортированный в алфавитном порядке по возрастанию. Стиль Summary (Сводный список) определяет, что должны отображаться названия различных факультетов и количество вакансий для каждого из них. Данный тип представления может быть очень удобен для отображения списков в виде каталогов. Кроме того, модуль Views позволяет динамически управлять отображением URL и названием.

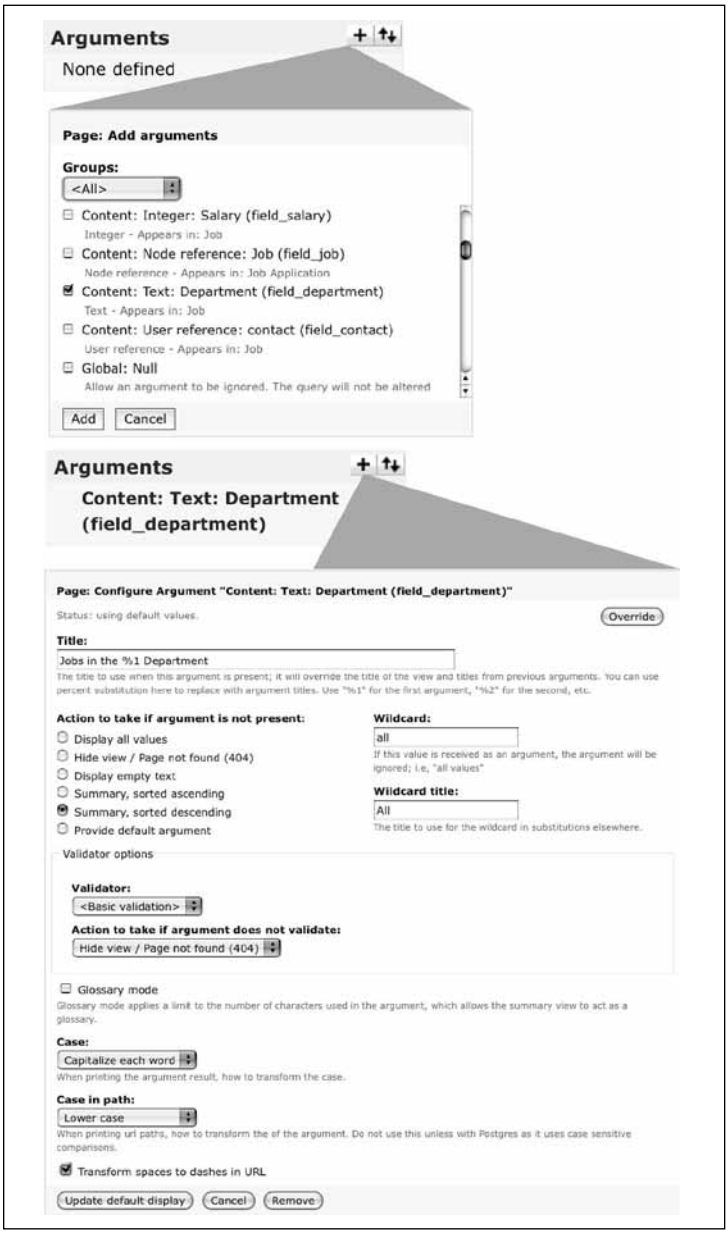


Рис. 3.33. *Настройки аргумента для представления Available Positions*

15. Щелкните на кнопке Update (Обновить), и вы получите серию параметров настройки аргумента. Оставьте значение по умолчанию для каждого из них, щелкнув на кнопке Update (Обновить).

По завершении не забудьте сохранить представление.

В наше представление вакансий мы добавили всего один аргумент. Однако вы можете добавить столько аргументов, сколько пожелаете. Более того, они даже не должны быть одного и того же типа. Таким способом можно создавать многостраничные структуры, каждая из которых представляет дополнительные подробности для элементов, которые должны появиться в списке.

На этом мы закончили создание нашего представления со списком вакансий! В окончательном виде страница с настройками должна выглядеть, как изображено на рис. 3.34. Откройте страницу с представлением, щелкнув на новой ссылке Available positions (Доступные вакансии) в блоке навигационного меню слева или щелкнув на ссылке View Page (Просмотреть страницу) в интерфейсе настройки модуля Views. После добавления аргумента наше представление будет отображать иерархическую структуру! И выглядеть, как изображено на рис. 3.35.

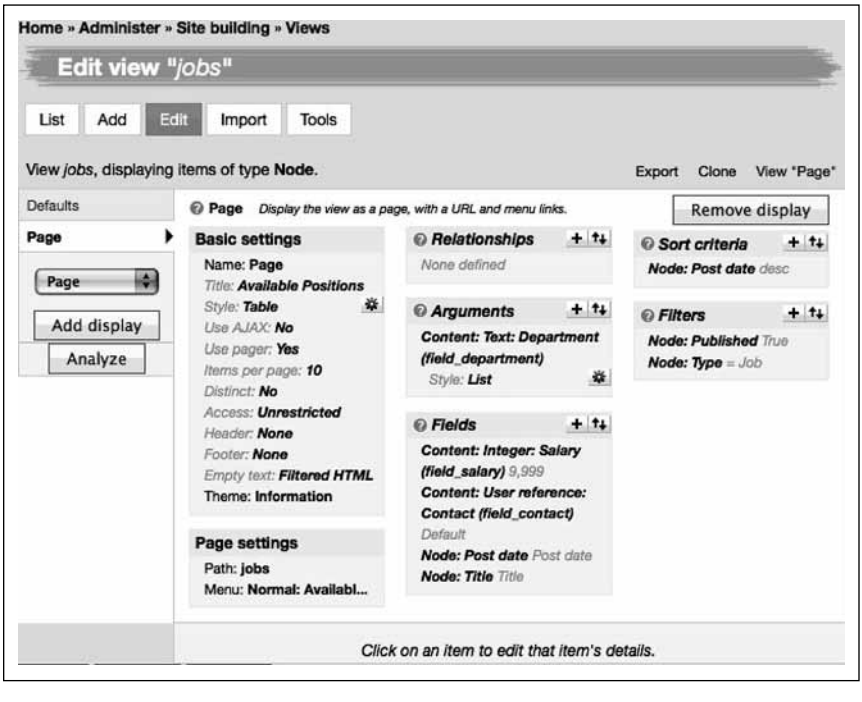


Рис. 3.34. *Окончательный вид страницы с настройками представления Available Jobs*

Щелчок на названии любого факультета приведет к отображению списка, отфильтрованного по этой категории, как показано на рис. 3.36. Выполняя переход между страницами, обратите также внимание на адрес URL. Он должен выглядеть примерно так: *http://www.example.com/jobs/administration* или *http://www.example.com/jobs/athletics*. Именно

так действуют аргументы в модуле Views: путь определяет адрес сводного списка, а за ним следуют какие-либо «подкаталоги» (такие как *administration*), которые интерпретируются как аргументы.

На этом мы завершаем разработку нашего первого представления, в котором мы использовали некоторые особенности модуля Views. В этом примере было создано единственное отображение (типа Page (Страница)), а кроме того, мы использовали простейший аргумент. В следующем примере мы создадим представление с несколькими отображениями и задействуем немного более сложные аргументы.

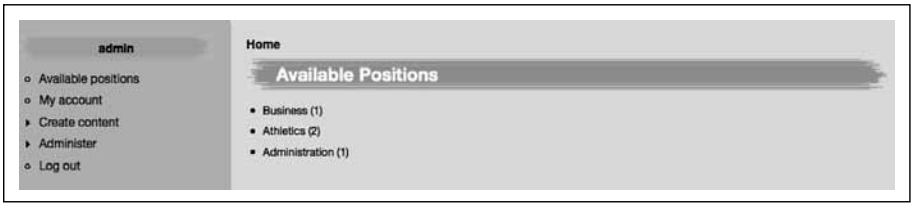


Рис. 3.35. *Самый верхний уровень представления Jobs после добавления аргумента*

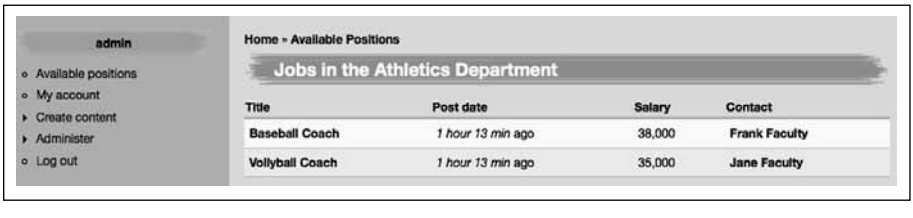


Рис. 3.36. *Отфильтрованное представление, которому был передан аргумент «athletics»*

Представление Job Applications (заявки)

Представление Job Applications будет играть роль инструмента для администраторов и ссылки – для пользователей. В нем будут реализованы следующие отображения:

- Список всех заявок на все вакансии в системе в виде единой страницы
- Список всех заявок для данной вакансии, отображаемый в виде вкладки на странице с объявлением о вакансии
- Список заявок, заполненных пользователями, в настоящее время присутствующими на сайте, отображаемый как блок в боковой колонке

После реализации всех отображений в окончательном виде представление должно выглядеть, как показано на рис. 3.37.

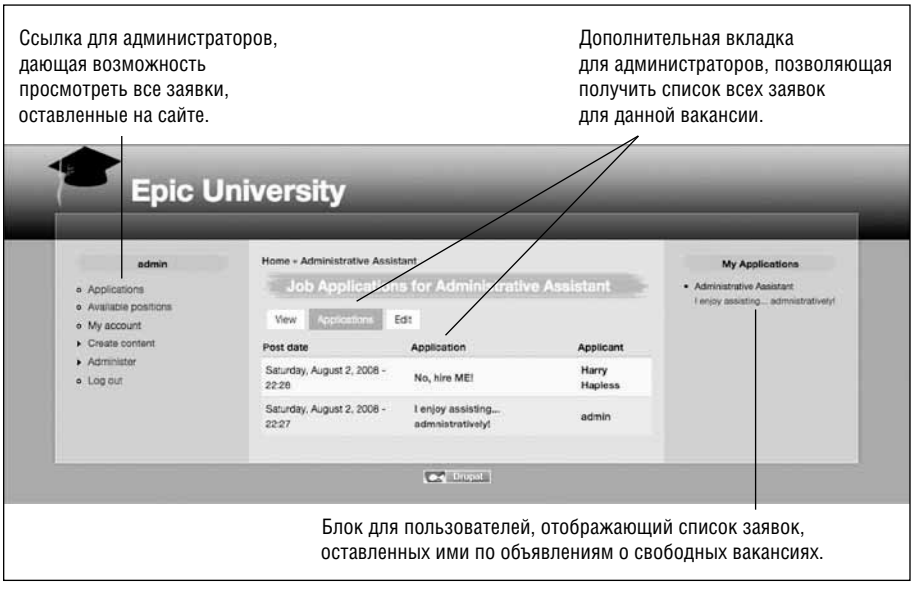


Рис. 3.37. *Несколько отображений представления Job Applications*

Создание представления и отображений по умолчанию

В этом представлении мы создадим несколько отображений, и затем внутри каждого из них будем переопределять значения параметров настройки по умолчанию. Определив большую часть настроек отображений, как значения по умолчанию, мы сэкономим массу времени позднее, когда нам потребуется изменять свойства, общие для всех отображений.

1. Перейдите на страницу администрирования модуля Views: Administer (Администрирование)→Site building (Конструкция сайта)→Views (*admin/build/views*). Щелкните на вкладке Add (Добавить), чтобы создать новое представление, определите значения параметров в соответствии с табл. 3.27 и затем щелкните на кнопке Next (Далее).

Таблица 3.27. Добавление настроек для представления applications

Поле	Значение
View name (Название представления)	applications
View description (Описание представления)	A list of submitted applications for a job or by a user (Список заявок для вакансии или для пользователя)
View tag (Рубрика представления)	jobs
View type (Тип представления)	Node (Материал)

2. Создав представление, установите значения параметров настройки представления в разделе Defaults (По умолчанию) в соответствии с табл. 3.28. По окончании щелкните на кнопке Update (Обновить).

Таблица 3.28. Базовые настройки представления applications

Defaults: Basic settings (базовые настройки)	**Значение**
Style (Стиль)	Table (Таблица)
Use pager (Использовать компонент выбора страниц)	Yes (Да) Full pager (Полнофункциональный компонент)
Empty text (Текст для пустого представления)	No applications yet. (Пока нет ни одной заявки.)

3. Теперь настроим каждое из полей по умолчанию, критерии сортировки и фильтры. Щелкните на ярлыке с изображением знака плюс в области Fields (Поля). Включите поля, описанные в табл. 3.29, и щелкните на кнопке Add (Добавить), затем определите значение каждого поля, щелкая на кнопке Update (Обновить) после каждого определения.

Таблица 3.29. Настройки полей по умолчанию для представления applications

Defaults: Configure field setting (по умолчанию: параметры настройки полей)	**Значение**
Content: Node reference: Job (Содержание: Ссылка на узел: Job)	Link this field to its node (Связать это поле с указанным узлом): отмечено
Node: Post date (Узел: Дата создания)	Date format (Формат даты): Sunday, May 4, 1078 - 05:00
Node: Title (Узел: Заголовок)	Label (Название): Application (Заявка) Link this field to its node (Связать это поле с указанным узлом): отмечено
User: Name (Пользователь: Имя)	Label (Название): Applicant (Соискатель)

4. Щелкните на ярлыке ↑↓ (обозначающем операцию переупорядочения) в разделе Fields (Поля) и расставьте поля в следующем порядке:
   - Node: Post date
   - Node: Title

- User: Name
- Content: Node Reference: Job

5. Щелкните на ярлыке с изображением знака плюс в области Sort criteria (Критерий сортировки) и добавьте критерий Post date (Дата создания). Настройте его в соответствии с табл. 3.30 и щелкните на кнопке Update (Обновить).

Таблица 3.30. Критерий сортировки по умолчанию для представления applications

Defaults: Sort criteria (критерий сортировки)	**Значение**
Node: Post date (Узел: Дата создания)	Sort order: Descending (Порядок сортировки: По убыванию)

6. Щелкните на ярлыке с изображением знака плюс в области Filters (Фильтры), отметьте фильтры Node: Published (Материал: Опубликовано) и Node: Type (Материал: Тип) и щелкните на кнопке Add (Добавить). Настройте фильтры в соответствии с табл. 3.31, щелкая на кнопке Update (Обновить) после настройки каждого фильтра.

Таблица 3.31. Фильтры по умолчанию для представления applications

Defaults: Filters (По умолчанию: Фильтры)	**Значение**
Node: Published (Узел: Опубликовано)	Published (Опубликовано): отмечено
Node: Type (Узел: Тип)	Operator (Оператор): Is one of (один из) Node type (Тип узла): Job Application

7. Сохраните настройки представления Applications, изображенные на рис. 3.38, щелчком на кнопке Save (Сохранить).

На этом заканчивается настройка значений по умолчанию для представления с заявками на вакансии. Оно пока нигде не отображается на сайте, но далее мы создадим простую страницу отображения с этими настройками по умолчанию.

Создание страницы отображения

В первом нашем отображении, которое выводит список всех заявок, оставленных на сайте, мы будем использовать настройки по умолчанию. Эта страница будет доступна только пользователям, обладающим ролью «editor», и мы создадим для нее пункт меню, который будет ото-

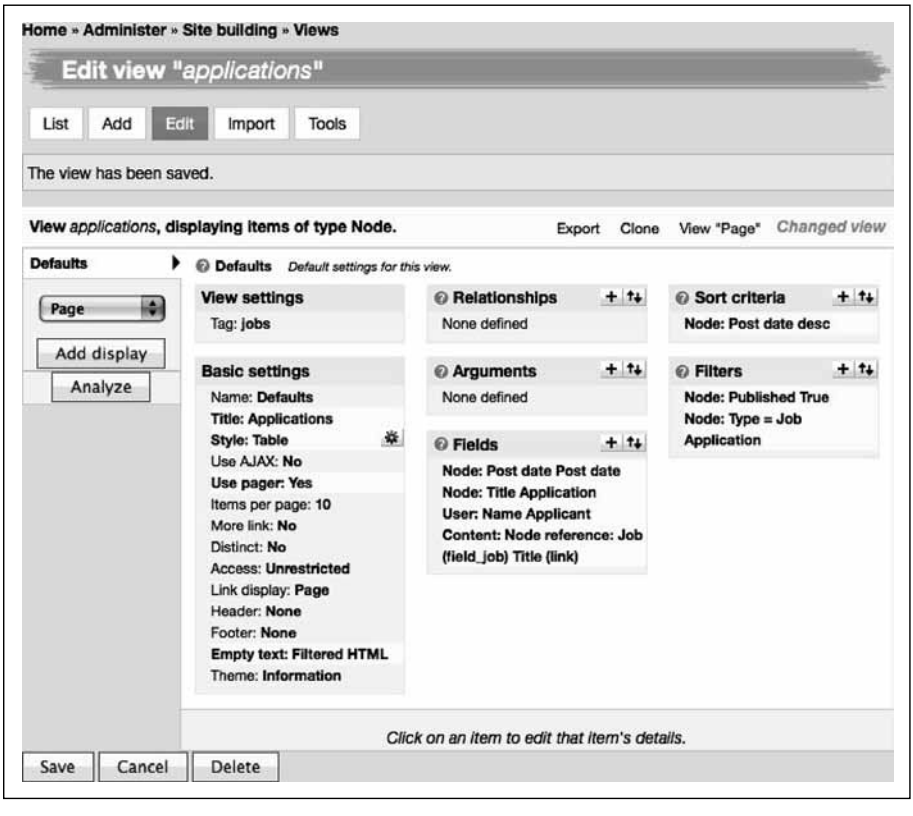


Рис. 3.38. *Вид страницы с настройками представления Job Applications к настоящему моменту*

бражаться в главном навигационном меню. По окончании страница с настройками будет выглядеть, как изображено на рис. 3.39.

Если вы уже покинули страницу настройки представления, вернитесь обратно: Administer (Администрирование)→Site building (Конструкция сайта)→Views→Applications→Edit (Изменить) (*admin/build/views/edit/applications*).

1. Выберите значение Page (Страница) в списке выбора типа отображения и щелкните на кнопке Add display (Добавить отображение).
2. Теперь переопределим первый параметр настройки. По умолчанию параметр Access (Доступ) для этого представления имеет значение Unrestricted (Не ограничен), однако нам требуется, чтобы доступ к полному списку заявок имели только пользователи, наделенные ролью «editor». Щелкните на параметре Access: Unrestricted (Доступ: Не ограничен) в разделе Basic Settings (Базовые настройки).

Applications

Post date	Application	Applicant	Job
Saturday, August 2, 2008 - 22:28	No, hire ME!	Harry Hapless	Administrative Assistant
Saturday, August 2, 2008 - 22:27	I enjoy assisting... admnistratively!	admin	Administrative Assistant
Saturday, August 2, 2008 - 22:27	I've always wanted to direct alumni	Joe Jobless	Alumni Director
Wednesday, July 30, 2008 - 22:38	Hire me now!	user	Alumni Director

Рис. 3.39. *Страница отображения со списком заявок в действии*

Обратите внимание, что для данного параметра сейчас используется значение по умолчанию. Щелкните на кнопке Override (Переопределить) в форме настройки этого параметра, задайте настройки в соответствии с табл. 3.32, чтобы ограничить доступ к этому отображению, и затем щелкните на кнопке Update (Обновить). По окончании настройки должны выглядеть, как изображено на рис. 3.40.

Таблица 3.32. Ограничение доступа к странице отображения для представления applications

Basic settings (базовые настройки)	**Значение**
Title (Название)	Щелкните на кнопке Override (Переопределить) Applications
Access (Доступ)	Щелкните на кнопке Override (Переопределить) Type (Тип): By role (по роли) If by role (Если имеет роль): editor

Ограничение доступа к представлению поможет предотвратить возможность обращения непривилегированных пользователей к отображению по адресу *http://www.example.com/applications*, но это не помешает им ввести адрес *http://www.example.com/node/4*, где узел с номером 4 – это заявка, которая им не принадлежит. Чтобы защититься от такого низкоуровневого способа доступа, необходимо задействовать модуль доступа к узлу, такой как Taxonomy Access Control (*http://drupal.org/project/taxonomy_access*) или Organic Groups (*http://drupal.org/project/og*).

3. Теперь настроим параметры отображения, чтобы присвоить этой странице собственный путь URL и пункт меню. Измените значения параметров в области Page settings (Настройки страницы) в соответствии с табл. 3.33 и щелкните на кнопке Update (Обновить).

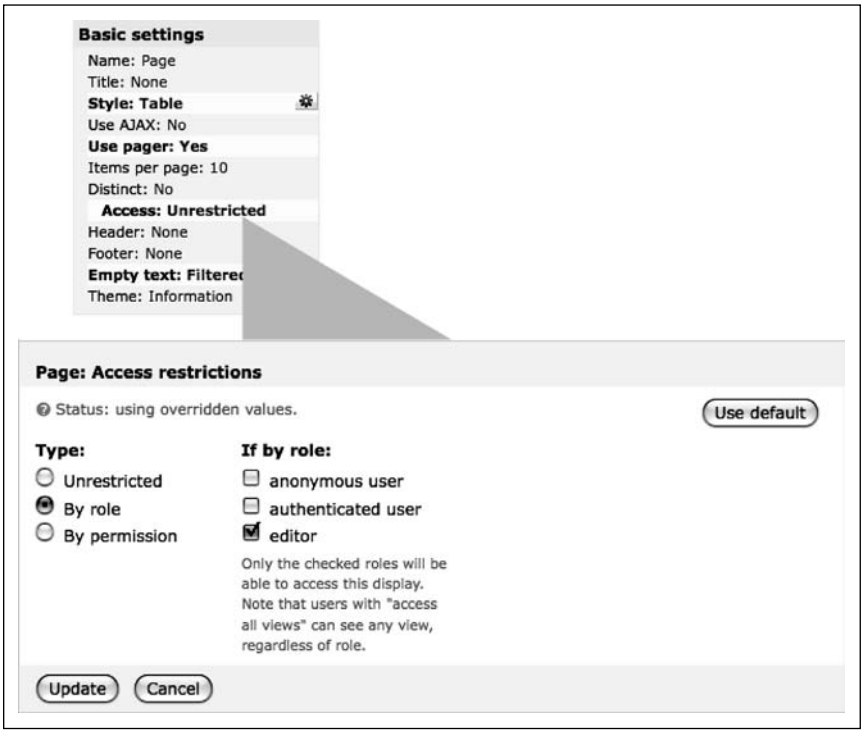


Рис. 3.40. *Переопределение отдельных базовых настроек в представлении*

Таблица 3.33. Настройки страницы отображения для представления applications

Page settings (настройки страницы)	**Значение**
Path (Путь)	applications
Menu (Меню)	Type (Тип): Normal menu entry (Обычный элемент меню) Title (Заголовок): Applications

4. Щелкните на кнопке Save (Сохранить), чтобы обновить изменения. Теперь настройки представления должны выглядеть, как изображено на рис. 3.41. Ссылка Applications (Заявки), появившаяся в главном навигационном меню сайта, будет вести на новую страницу отображения.

Создание отображения Job tab

Мы уже создали страницу, содержащую список всех заявок, оставленных на сайте. Это может быть удобно для тех, кто просматривает посту-

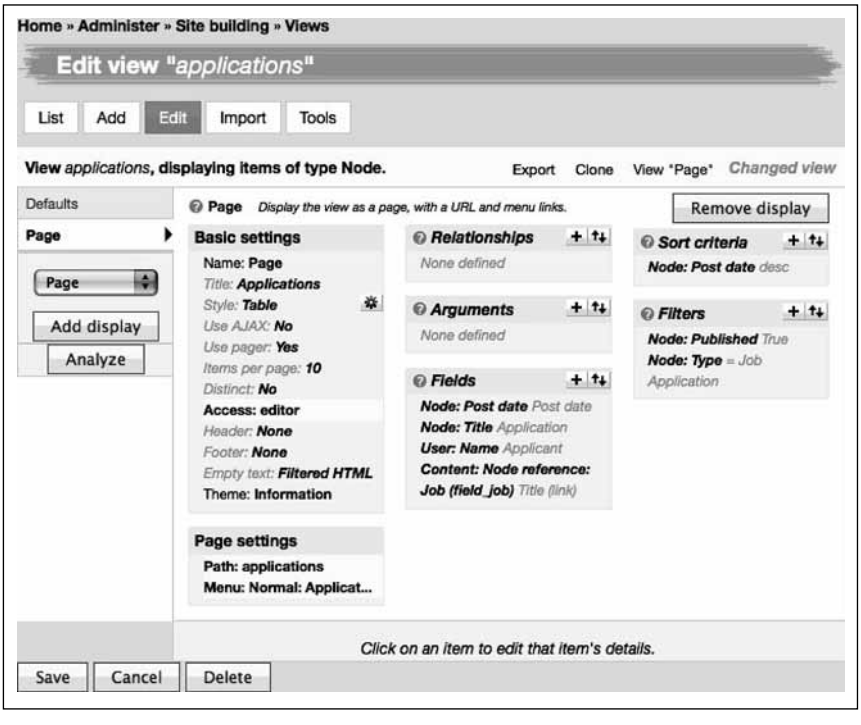


Рис. 3.41. *Представление Job Applications с отображением типа Page*

пающие заявки, однако это неудобно для пользователей, которых интересуют заявки, оставленные для какой-то конкретной вакансии. Чтобы удовлетворить эту потребность, мы создадим отображение, в котором будет представлена только одна вакансия, заданная аргументом. Для удобства поиска этой страницы мы добавим ее как вкладку на страницы с объявлениями о вакансиях (рис. 3.42).

Рис. 3.42. *Представление Applications отображается как вкладка на странице узла типа Job*

Для достижения поставленной цели нам потребуется задействовать новую особенность модуля Views, которую мы еще не использовали: *от-*

ношения. Получить список заявок на вакансии несложно, но в данном случае нам требуется получить информацию из того представления Job, которое соответствует отображаемой вкладке, а именно: нам необходим идентификационный номер узла этого представления, чтобы определить, какие заявки отображать. Для извлечения данных из представления типа Job мы можем использовать отношение и таким образом использовать его для работы с заявками. На рис. 3.43 показано, как действуют отношения.

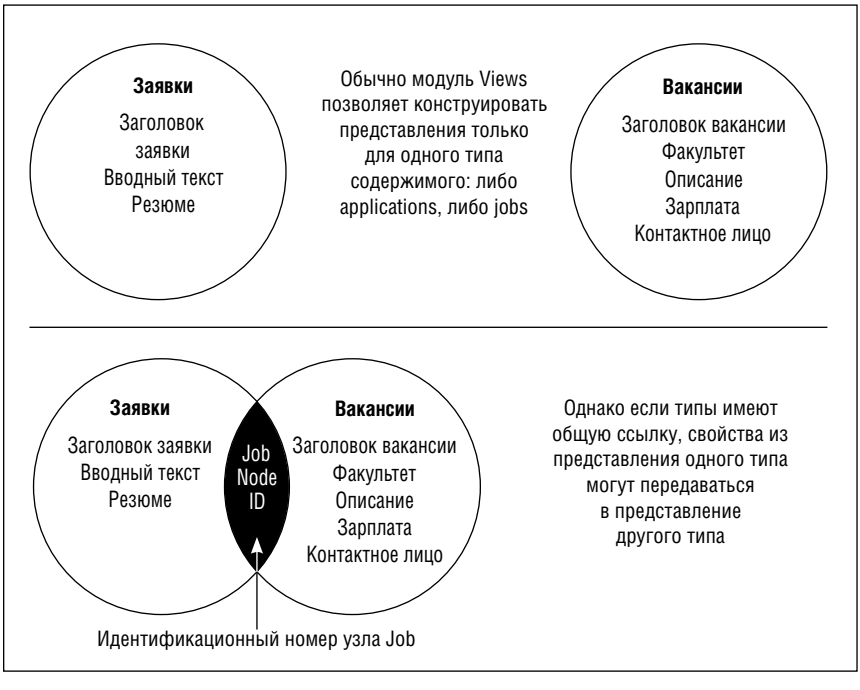


Рис. 3.43. *При наличии общей связи отношения позволяют передавать данные из одного представления в другое*

Если вы уже покинули страницу администрирования модуля Views, вернитесь на нее: Administer (Администрирование)→Site building (Конструкция сайта)→Views→Applications→Edit (Изменить) (*admin/build/views/edit/applications*).

1. Выберите тип отображения Page (Страница) в раскрывающемся списке select и щелкните на кнопке Add display (Добавить отображение).
2. Новое отображение по умолчанию получит имя Page. Чтобы как-то отличать его от типа отображения Page, переименуем его в Job Tab, так как эта страница будет отображаться как вкладка в узлах типа Job. Кроме того, переопределим права доступа к отображению, чтобы его могли просматривать только пользователи с ролью «editor».

Измените настройки в разделе Basic settings (Базовые настройки) этого отображения в соответствии с табл. 3.34.

Таблица 3.34. Базовые настройки отображения Job Tab для представления applications

Job Tab: Basic settings (базовые настройки)	**Значение**
Name (Имя)	Job Tab
Access (Доступ)	Щелкните на кнопке Override (Переопределить) Type (Тип): By role (по роли) Role (Роль): editor

3. Добавьте отношение, щелкнув на ярлыке с изображением знака плюс в области Relationships (Отношения), установите значения параметров настройки в соответствии с табл. 3.35 (и как показано на рис. 3.44) и щелкните на кнопке Update default display (Обновить отображение по умолчанию).

Таблица 3.35. Отношения по умолчанию для представления applications

Defaults: Relationships (По умолчанию: Отношения)	**Значение**
Content: Node reference: Job (Материал: Ссылка на узел: Job)	Require this relationship (Требовать это отношение): отмечено

4. Подобно представлению jobs, которое мы настраивали выше, мы можем отфильтровать список заявок, добавив в отображение аргумент. Это первый случай, когда мы переопределяем часть представления, отличную от параметров настройки. В данном случае переопределение выполняется несколько иначе. Щелкните на заголовке раздела Arguments (Аргументы); это позволит переопределить все аргументы представления за один раз. Нет никакой возможности переопределить только один аргумент – отдельное отношение, поле, фильтр или критерий сортировки. Когда часть представления будет переопределена, значения параметров будут выводиться некурсивным шрифтом, показывая тем самым переопределенные значения, как изображено на рис. 3.45.

   Приступив к переопределению аргументов отображения Job Tab, мы добавим аргумент, который фильтрует список, оставляя в нем заявки только для одной вакансии. Щелкните на ярлыке с изображением знака плюс в области Arguments (Аргументы) и определите значения в соответствии с табл. 3.36. Аргумент Node ID (Идентификатор узла) позволяет фильтровать заявки по определенному номеру узла

с информацией о вакансии. Здесь мы задействовали еще одну особенность аргументов модуля Views: *валидатор*. Модуль Views будет проверять идентификационный номер узла, переданный в строке адреса URL, на принадлежность узлу с информацией о вакансии. По окончании щелкните на кнопке Update (Обновить).

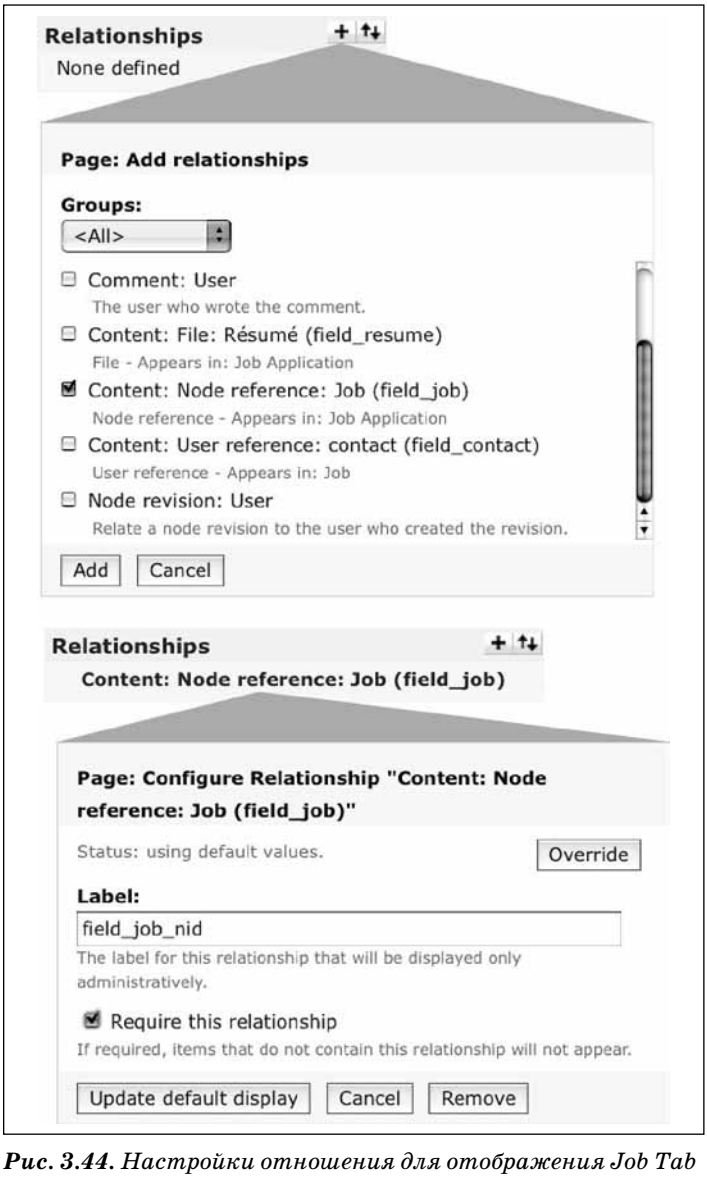


Рис. 3.44. *Настройки отношения для отображения Job Tab*

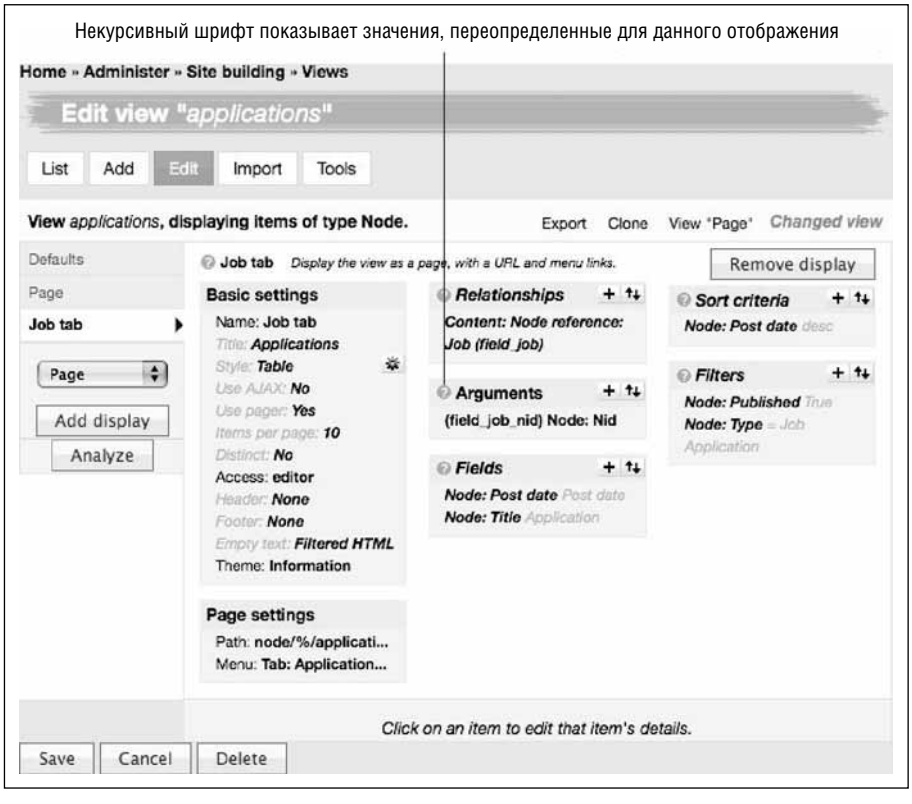


Рис. 3.45. *Переопределение настроек представления в разделе Arguments (Аргументы)*

Таблица 3.36. Аргументы отображения Job Tab представления applications

Job Tab: Настройки аргументов	**Значение**
Node: Nid (Узел: Nid)	Relationship (Отношение): field_job_nid Title (Заголовок): Job Applications for %1 (Заявки для вакансии %1) Action to take if argument is not present: Display empty text (Действие, выполняемое в случае отсутствия аргумента: Отображать текст для пустого представления) Validator (Валидатор): Node (Узел) Types (Типы): Job Validate user has access to the node (Проверять права доступа пользователя к узлу): отмечено

5. Теперь, когда мы настроили аргумент для данного отображения, нам необходимо передать ему адрес URL. Измените настройки в разделе Page settings (Настройки страницы) в соответствии с табл. 3.37 и щелкните на кнопке Update (Обновить).

Таблица 3.37. Настройки страницы для отображения Job Tab

Job Tab: Page settings (Настройки страницы)	**Значение**
Path (Путь)	node/%/applications
Menu (Меню)	Type (Тип): Menu tab (Вкладка меню) Title (Заголовок): Applications (Заявки)

Аналогично размещению %1 в случае заголовка, размещением знака процента мы определяем положение первого аргумента в середине адреса URL. Этот подход можно также использовать для добавления вкладок на страницы пользователей, например `user/%/my_display`, или на любые другие страницы в Drupal, где можно задавать пути динамически.

6. И наконец, нам больше не нужен список вакансий в этом отображении; он стал лишним, так как мы будем просматривать саму вакансию. Щелкните на заголовке раздела Fields (Поля) и затем на кнопке Override (Переопределить). Затем щелкните на ссылке Content: Node reference: Job (Содержимое: Ссылка на узел: Job) и на кнопке Remove (Удалить).

Если требуется определить настройки, относящиеся к какому-то одному отображению и не относящиеся к остальным, не забудьте щелкнуть на кнопке Override (Переопределить). Настройки по умолчанию модуля Views в разделе Defaults display (Настройки отображения по умолчанию) определяют поведение всех отображений в этом представлении. Если не перейти в режим переопределения, можно случайно удалить ту или иную настройку сразу из нескольких отображений.

7. Щелкните на кнопке Save (Сохранить), чтобы сохранить представление, настройки которого теперь должны выглядеть, как показано на рис. 3.46.

Теперь мы добавили вкладку с заявками во все узлы с информацией о вакансиях (для пользователей, наделенных ролью «editor»). Откройте узел вакансии, для которой имеются заявки; он должен выглядеть примерно так, как уже было изображено на рис. 3.41.

Создание блока для отображения Applications (Заявки)

Последнее отображение, которое мы сконструируем, будет доступно всем пользователям сайта. Это будет блок, отображающий все заяв-

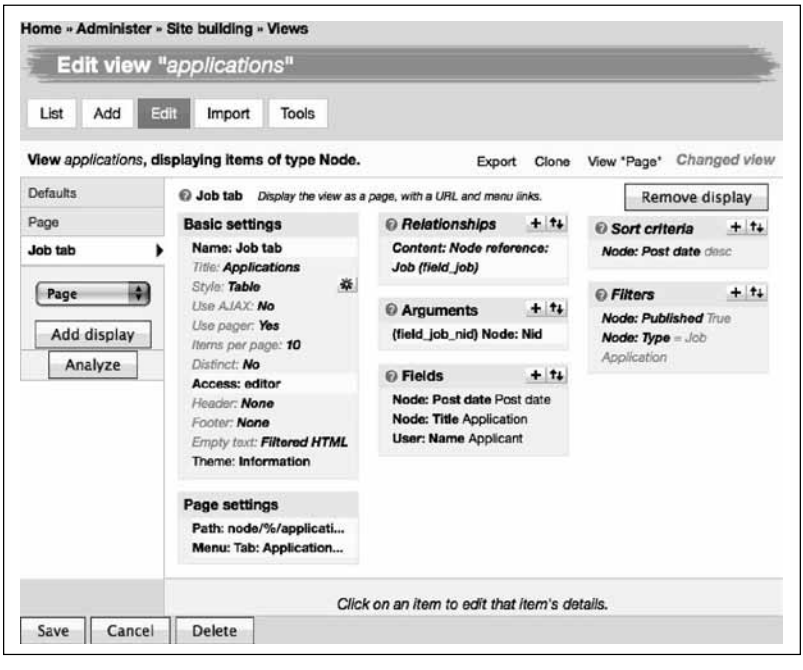


Рис. 3.46. *Представление Applications с новым отображением Job tab*

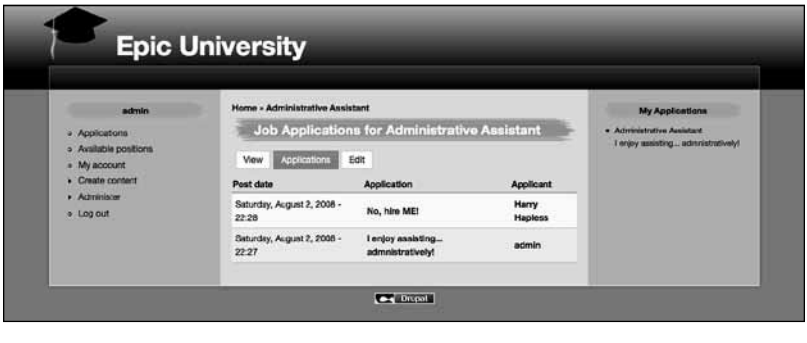


Рис. 3.47. *Представление Applications с отображением User Applications в виде блока в правой колонке*

ки, принадлежащие текущему пользователю, находящемуся на сайте. Кроме того, мы изменим стиль отображения – теперь заявки будут выводиться не в виде таблицы, а в виде списка, потому что его придется отображать в более узкой боковой колонке. Окончательный вид отображения показан на рис. 3.47.

Вернитесь на страницу настройки представления Applications: Administer (Администрирование)→Site building (Конструкция сайта)→Views→Applications→Edit(Изменить)(*admin/build/views/edit/applications*). Начнем с добавления в это представление нового отображения:

1. Выберите в раскрывающемся списке тип отображения Block (Блок) и щелкните на кнопке Add display (Добавить отображение).
2. Переопределите базовые настройки отображения в соответствии с табл. 3.38 и после изменения каждого параметра щелкните на кнопке Update (Обновить).

Таблица 3.38. Базовые настройки отображения Application Block для представления Applications

Block: Basic settings (Базовые настройки)	**Значение**
Title (Название)	Щелкните на кнопке Override (Переопределить) My applications (Мои заявки)
Style (стиль)	Щелкните на кнопке Override (Переопределить) Style (Стиль): List (Список) Type (Тип): Unordered list (Неупорядоченный список)
Row style (Стиль строки)	Fields (Поля)

3. Теперь определим текст описания блока, которое будет отображаться в области управления блоками. В разделе Block settings (Настройки блока) введите текст «User Applications» (Заявки пользователя) в параметр Admin (Для администратора) и щелкните на кнопке Update (Обновить).

Теперь мы по сути определили три имени, или заголовка, для этого блока. Ниже описывается, где будет отображаться каждый из заголовков:

Basic settings: Name (Базовые настройки: Имя)

Используется как имя отображения в интерфейсе настройки представления. Отображается в виде вкладки в левой части интерфейса.

Basic settings: Title (Базовые настройки: Заголовок)

Используется как заголовок блока при отображении перед конечным пользователем.

Block settings: Admin (Настройки блока: Для администратора)

Используется для обозначения блока в списке на странице Administer (Администрирование)→Site building (Конструкция сайта)→Blocks (Блоки) (*admin/build/block*).

1. Поскольку данное отображение является блоком, который располагается в боковой колонке, желательно было бы уменьшить количество отображаемых полей, чтобы блок аккуратно вписывался

в узкую область на странице. Для этого нужно переопределить настройки этого представления в области Fields (Поля). Щелкните на заголовке раздела Fields (Поля) и затем на кнопке Override (Переопределить) в форме настройки.

2. Теперь, когда был выбран режим переопределения полей для отображения Applications Block, удалите поля Node: Post date (Узел: Дата создания) и User: Name (Узел: Имя). Для этого нужно сначала щелкнуть на ссылке поля, а затем на кнопке Remove (Удалить).

Вы можете быстро удалить сразу несколько полей, щелкнув на ярлыке с изображением ↑↓ (обозначающем операцию переупорядочения) и затем щелкнув на ссылке remove (удалить) рядом с нежелательными полями.

3. Переименуйте поля в соответствии с табл. 3.39.

Таблица 3.39. Поля отображения Applications Block в представлении Applications

Block: Fields (Блок: Поля)	Значение
Node: Title (Узел: Заголовок)	Label (Название): (пустое)
Content: Node reference: Job (Содержание: Ссылка на узел: Job)	Format (Формат): Title (no link) (Заголовок (без ссылки)) Link this field to its node (Связать это поле с указанным узлом): не отмечено Label (Название): (пустое)

4. Измените порядок следования полей, используя кнопку ↑↓ (обозначающую операцию переупорядочения), чтобы первым следовало поле с заголовком вакансии, а затем с заголовком заявки.
5. Наконец, так как в этом отображении требуется выводить заявки, принадлежащие только текущему пользователю, мы должны добавить фильтр в отображение. Как и в случае с полями, чтобы добавить новый параметр, необходимо переопределить все фильтры. Щелкните на заголовке раздела Filters (Фильтры) и затем на кнопке Override (Переопределить) в области с настройками.
6. Два существующих фильтра (Node: Published (Узел: Опубликовано) и Node: Type (Узел: Тип)) оставим без изменений и добавим новый фильтр для параметра User: Current (Пользователь: Текущий). Включите этот фильтр, Щелкните на кнопке Add (Добавить), отметьте флажок Is the logged-in user (Пользователь, присутствующий на сайте) и щелкните на кнопке Update (Обновить). Этот шаг позволит динамически изменять содержимое блока в зависимости от текущего пользователя, присутствующего на сайте.

7. Сохраните представление. Теперь настройки представления должны выглядеть, как изображено на рис. 3.48. На этом мы закончили настройку блока отображения User Applications.
8. Добавив новое отображение типа Block (Блок), мы добавили новый блок в экземпляр сайта на базе Drupal. Прежде чем это отображение станет видимым, необходимо блок включить. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Blocks (Блоки) (*admin/build/block*) и включите новый блок. Поместите его в область Right sidebar (Правая колонка) и сохраните изменения.

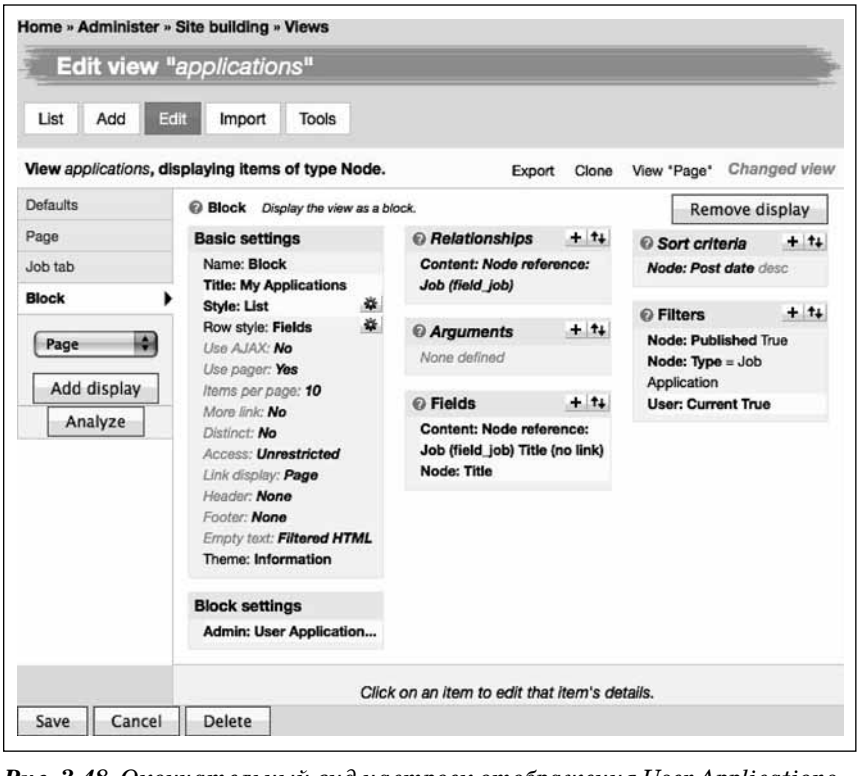


Рис. 3.48. *Окончательный вид настроек отображения User Applications для представления Applications*

Дополнительно

Простой веб-сайт каталога вакансий, который мы только что сконструировали, использует лишь некоторые из возможностей модулей CCK и Views. Существует еще множество возможностей расширения функциональности этого сайта за счет добавления дополнительных полей в типы содержимого Job и Job Application. Ниже приводится несколь-

ко модулей, которые можно было бы использовать для расширения возможностей пакета CCK:

Automatic Node Titles, http://drupal.org/project/auto_nodetitle

Этот модуль обеспечивает поддержку шаблонов заголовков для узлов. Например, вместо того чтобы предоставлять пользователям возможность для своих заявок вручную вводить заголовки (которые могут оказаться довольно бессмысленными, например, «Возьмите меня!»), с помощью этого модуля можно автоматически обеспечить соответствие заголовков всех заявок стандартному формату, такому как [author-name] – [job-title].

Custom Links, http://drupal.org/project/custom_links

Этот удобный модуль может использоваться для добавления в любой узел в дополнение к обычным ссылкам ваших собственных ссылок, таких как «Добавить новый комментарий». Эту возможность мы могли бы использовать для добавления прямых ссылок на страницу создания заявки непосредственно из страницы с описанием вакансии, чтобы пользователям не нужно было выбирать требуемую вакансию из раскрывающегося списка в форме создания заявки.

Content Permissions (включается в CCK)

Если возникает необходимость ограничить доступ непривилегированных пользователей к частной информации, содержащейся в заявке, то с помощью модуля Content Permissions можно скрыть некоторые поля в зависимости от роли пользователя. Так как этот модуль включается в состав CCK, отпадает необходимость загружать и устанавливать его отдельно.

В заключение

Эта глава рассказала, как использовать два фундаментальных «строительных блока» системы Drupal: модули CCK и Views. Эти модули обеспечивают значительную мощь Drupal и будут широко использоваться на протяжении последующих глав. Пакет CCK используется для моделирования содержимого веб-сайта, позволяя добавлять дополнительные поля с целью образования различных свойств, а модуль Views используется для отображения списков данных, имеющихся на веб-сайте.

Помимо основных особенностей перечисленных модулей в этой главе была представлена методология конструирования сайта на базе платформы Drupal. В системе Drupal вместо использования монолитных пакетов каждый модуль реализует определенный круг функциональных возможностей и работает совместно с другими модулями, расширяя их функциональность. Поскольку мы создавали дополнительные поля для различных типов содержимого, пакет CCK работал совместно с базовым модулем Node. В процессе создания списков содержимого модуль Views извлекал информацию и из базовых модулей, и из пакета CCK. Тако-

го рода взаимодействия между модулями являются фундаментальной концепцией для остальной части книги, так как все больше модулей будет привлекаться к работе, придавая новые оттенки нашим сайтам.

Ниже приводится список модулей, использованных в этой главе:

- Модуль Automatic Nodetitle: *http://drupal.org/project/auto_nodetitle*
- Пакет Content Construction Kit (CCK): *http://drupal.org/project/cck*
- Модуль Custom Links: *http://drupal.org/project/custom_links*
- Модуль FileField: *http://drupal.org/project/filefield*
- Модуль Link: *http://drupal.org/project/link*
- Модуль Token: *http://drupal.org/project/token*
- Модуль Views: *http://drupal.org/project/views*

Ниже приводится список ресурсов, упоминавшихся в этой главе, и ресурсов сообщества для более подробного изучения новых концепций, представленных в этой главе:

- Группа разработчиков пакета CCK (CCK Developers Drupal group): *http://groups.drupal.org/cck*
- Модули реализации полей для пакета CCK: *http://drupal.org/project/Modules/category/88*
- Группа разработчиков модуля Views (Views Developers Drupal group): *http://groups.drupal.org/views-developers*

4

Обзоры продуктов

По мере того как с каждым днем в Интернете появляются все новые и новые предложения для покупателей, правильный выбор товара может стать серьезной проблемой. Специализированные веб-сайты, представляющие описания характеристик отдельных видов товаров и обзоры экспертов, пользуются большой популярностью у потребителей, помогая им находить нужные товары среди многообразия выбора. В этой главе мы с помощью полезных модулей Drupal сконструируем веб-сайт каталога товаров, который позволит членам сообщества высказывать свои мнения о каждом представленном продукте.

В этой главе будут представлены следующие модули:

Amazon (http://drupal.org/project/amazon)

Собирает информацию о продуктах с сайта Amazon.com (*http://www.amazon.com*)

Voting API (http://drupal.org/project/votingapi)

Обеспечивает платформу для проведения голосований

Fivestar (http://drupal.org/project/fivestar)

Реализует систему рейтинговых оценок

Search (базовый)

Индексирует содержимое сайта и реализует функцию поиска

CSS Injector (http://drupal.org/project/css_injector)

Обеспечивает для администраторов простоту добавления стилей CSS

Если вы собираетесь следовать за практическими упражнениями этой главы, вам следует установить Drupal, используя профиль установки Reviews из примеров кода, распространяемых вместе с книгой. В результате применения этого профиля на вашем веб-сервере будет создана заготовка веб-сайта. В окончательном виде этот веб-сайт будет вы-

глядеть, как показано на рис. 4.1 и по адресу *http://reviews.usingdrupal.com*. За дополнительной информацией об использовании профилей установки обращайтесь к разделу «Вступление».

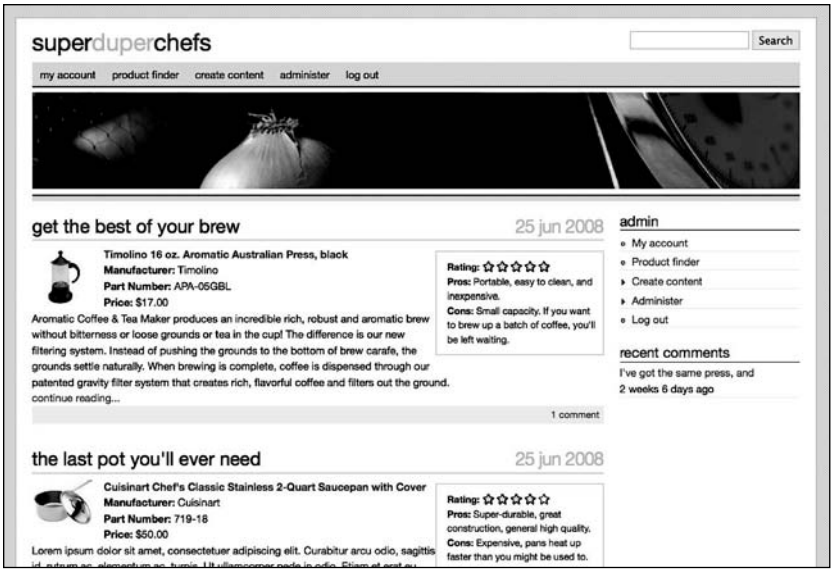


Рис. 4.1. *Окончательный вид веб-сайта Super Duper Chefs*

Задача

Боб и Сара работают вместе и увлекаются приготовлением кулинарных изысков. Они оба обставили свои кухни по последнему слову кулинарной мысли и наполнили их всевозможными приспособлениями, посудой и техникой для приготовления пищи. Однако следование за кулинарной модой может оказаться достаточно дорогим удовольствием, поэтому, прежде чем купить новый прибор, они часто обращаются за советом к гурманам. Они решили создать свой веб-сайт Super Duper Chefs (Супер-пупер повара), где они и их друзья могли бы оставлять отзывы о своем кухонном оборудовании, делиться советами и хвастаться своими последними кулинарными достижениями. Они хотят создать такой сайт, какого им не хватало, когда только начинали: привлекательное для многих место, где рассказывается о самых полезных приспособлениях и даются разнообразные советы.

Обсудив идею со своими друзьями, Боб и Сара поняли, какую информацию должен предлагать сайт. Самое главное – обзоры товаров для кухни, предоставленные зарегистрированными обозревателями, должны содержать рейтинговую оценку и краткий перечень их достоинств и недостатков. Кроме того, каждое описание должно содержать самую свежую информацию о цене. В дополнение к официальным обзорам по-

сетители сайта должны иметь возможность высказывать собственное мнение о товарах и сравнивать официальные оценки с мнениями других посетителей, прочитавших обзор. Любой посетитель сайта должен также иметь возможность отыскивать описания, соответствующие определенным критериям. Например, посетитель должен без труда находить обзоры товаров конкретного производителя или товары по неточному названию.

Примечания к реализации

Следующий шаг заключается в том, чтобы выяснить, как перевести этот набор требований в функциональные возможности Drupal. Бобу и Саре повезло: средств ядра системы Drupal достаточно, чтобы удовлетворить большую часть из того, что им требуется. Мы дадим обозревателям право с помощью инструментов администрирования Drupal создавать содержимое типа Product Review (Обзор продукта). Эти обзоры составят основу информационного наполнения сайта.

Для добавления дополнительных полей в тип содержимого Product Review, где будут храниться различные сведения, мы задействуем пакет CCK. Мы также будем использовать модуль Views, с помощью которого сконструируем страницы со списками продуктов для быстрого ознакомления. Однако для удовлетворения требований, обусловленных создаваемым сайтом, нам потребуется функциональность, с которой мы еще не встречались: импортирование информации о продуктах с другого веб-сайта, обеспечение для пользователей возможности оценивать и комментировать содержимое и реализация страницы поиска по сайту.

Информация о продукте

В первую очередь сайт должен отображать информацию о рассматриваемых продуктах. Кто его производит? Сколько он стоит? Где посетитель веб-сайта может приобрести его? Для хранения информации, содержащей ответы на все эти вопросы, можно было бы создать специализированные поля с помощью модуля CCK, но для редакторов сайта заполнение всех этих полей для каждого обзора превратилось бы в муку. Более того, по мере роста сайта все труднее и труднее будет обеспечивать свежесть информации о стоимости.

Самое простое решение заключается в том, чтобы переложить эту работу на кого-нибудь другого! Сайт Amazon.com предоставляет доступ к своей базе данных, заполненной информацией о товарах, включая и кухонное оборудование (как видно на рис. 4.2), с использованием Amazon Associates Web Services API (прикладной программный интерфейс доступа к веб-службам компании Amazon Associates). Модуль Amazon (*http://drupal.org/project/amazon*) дает сайтам возможность получать эту информацию о продуктах. Это означает, что при обращении на сайт можно просто заполнить единственное поле с названием продукта, а все остальное будет сделано без вашего участия.

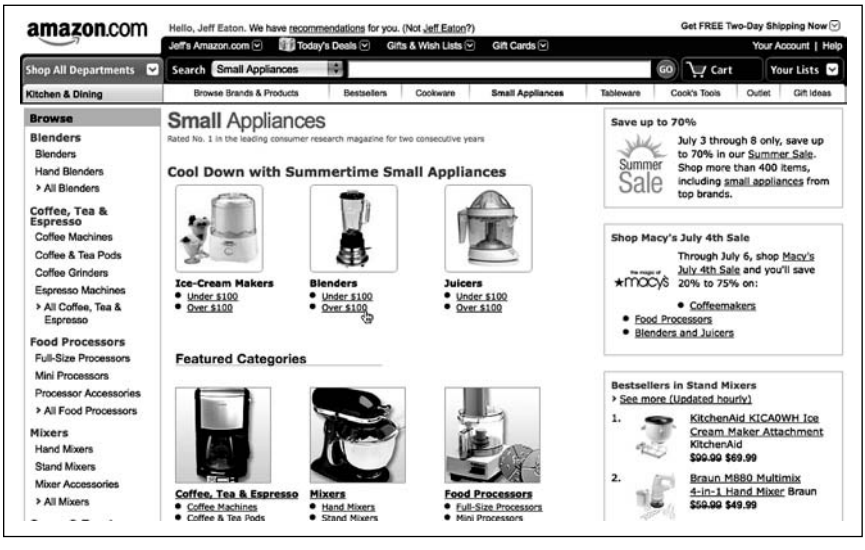


Рис. 4.2. *Оборудование для кухни на веб-сайте Amazon.com*

В дополнение к экономии времени и сил получение информации о продуктах с помощью Amazon API означает, что Боб и Сара смогут получать вознаграждение за перенаправление, когда кто-нибудь будет щелкать на их веб-сайте и приобретать что-нибудь на Amazon.com. Это простейший способ зарабатывать деньги, а для сайтов с высокой посещаемостью сумма комиссионных может расти очень быстро.

Оценка продуктов

Вторая проблема – оценка продуктов. Для каждого продукта, представленного на сайте, должен иметься официальный обзор, однако посетителям сайта должна быть предоставлена возможность оценить эти продукты. Наличие официальной оценки и, кроме того, оценки пользователей поможет повысить уровень доверия к представлению продукта и дать новым посетителям уверенность, что оценки, представленные на сайте, не во власти редакторов.

Сообществом Drupal были созданы десятки дополнительных модулей, позволяющих организовать на сайте голосование и выставление рейтинговых оценок. Полный перечень этих модулей можно найти по адресу *http://drupal.org/project/Modules/category/60*. Одни из них, такие как модуль NodeReview (*http://drupal.org/project/nodereview*), дают посетителям возможность оценивать содержимое сразу по нескольким критериям. Другие, такие как модуль Plus1 (*http://drupal.org/project/plus1*), дают возможность голосовать за элементы, постоянно наращивая их рейтинг, как на популярных сайтах Reddit (*http://www.reddit.com*) и Digg (*http://digg.com*). Имеются и такие, которые позволяют каждому читателю выставить оценку в определенном диапазоне и при каж-

дом новом посещении отображают среднее арифметическое значение этих оценок. Поскольку нас интересует возможность определения такой средней оценки, мы будем использовать для голосования модуль Fivestar (*http://drupal.org/project/fivestar*).

Помимо того, что модуль Fivestar дает пользователям возможность голосовать за материалы на сайте, он также добавляет поле CCK, позволяющее отделить «официальную оценку» автора материала от оценки, выставленной посетителями. Мы можем использовать модуль Views для вывода этих двух оценок рядом друг с другом, чтобы иметь возможность сравнивать их. Модуль Fivestar подобно большинству модулей голосования и оценки опирается на возможности, реализованные в Voting API (*http://drupal.org/project/votingapi*), еще одном модуле Drupal, который обеспечивает хранение и отображение результатов голосования за материалы. Нам необходимо установить его, чтобы получить возможность использовать модуль Fivestar.

Реализация поиска

Третий компонент нашей мозаики – это страница поиска, которая позволит посетителям сайта отыскивать необходимые им описания продуктов. Модуль Search, встроенный в Drupal, может индексировать содержимое каждого материала и предоставлять в распоряжение посетителей универсальную страницу поиска материалов, содержащих характерные ключевые слова. Однако этот модуль сложен в настройке представления результатов и в определении типов материалов, по которым следует выполнять поиск. Например, организация поиска обзоров кухонных приборов, написанных Бобом, и сортировка их по стоимости может оказаться далеко не простым делом. К счастью, модуль Views дает нам возможность использовать поисковые индексы и обеспечить полный контроль над отображением результатов. Поэтому мы обратимся к нему при конструировании нашей страницы поиска.

Первый шаг: основные описания продуктов

Прежде чем приступить к работе, зарегистрируйтесь на сайте как пользователь *admin* с паролем *oreilly*, если вы использовали профиль установки. Мы начнем с некоторых определений, которые нам потребуются в дальнейшем.

Первое, что нам потребуется, – это новый тип содержимого для описаний продуктов. Согласно требованиям, предъявляемым к сайту Super Duper Chefs, для каждого описания нам потребуются:

- Поля Pros (За) и Cons (Против), которые будут хранить краткие перечни достоинств и недостатков описываемых продуктов
- Поле Amazon Product (Продукт на Amazon), где будет храниться подробная информация о продукте

- Поле Rating (Оценка), чтобы пользователи могли быстро находить наилучшую оценку
- Поле Group (Группа), объединяющее оценку, а также все «за» и «против» для более эффективного представления информации
- Комментарии, чтобы посетители могли составить свое собственное мнение

Создание типа содержимого Product Review

Начнем с создания базового типа содержимого и добавления элементов: простых текстовых полей Pros (За) и Cons (Против), объединения этих полей и включения комментариев.

1. Сначала перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Modules (Модули) (*admin/build/modules*) и включите следующие модули:
   - CCK
     – Content
     – Fieldgroup
     – Text
2. Далее перейдите на страницу Administer (Администрирование)→Content management (Содержание)→Content types (Типы материалов) (*admin/content/types*) и добавьте новый тип содержимого с названием Product review (Обзор продукта) и определите настройки в соответствии с табл. 4.1.

Таблица 4.1. Настройки типа содержимого Product Review

Параметр настройки	Значение
Identification (Идентификация)	
Name (Имя)	Product review (Обзор продукта)
Type (Тип)	review
Description (Описание)	A featured product review by a contributing editor (Обзор продукта, представленный редактором)
Submission form settings (Свойства формы)	
Title field label (Название поля заголовка)	Headline (Заголовок)
Body field label (Название поля текста)	Review (Описание)

Comment settings (Настройки комментариев)	
Default comment setting (Настройки комментариев по умолчанию)	Read/Write (Чтение/Запись)
Comment subject field (Поле темы комментария)	Disabled (Отключено)
Location of comment submission form (Расположение формы отправки комментариев)	Display below post or comments (Показывать ниже сообщения или комментариев)

3. Сохранив изменения щелчком на кнопке Save content type (Сохранить тип содержимого), вы вернетесь к списку доступных типов содержимого. Щелкните на ссылке manage fields (управлять полями) (*admin/content/node-type/review/fields*) для типа содержимого Product review (Обзор продукта), чтобы приступить к настройке полей.
4. Так как в этом типе содержимого нам необходимо сгруппировать вместе несколько полей (Pros (За) и Cons (Против), оценку и другие), создадим сначала группу Field (Поле). Щелкнув на ссылке New group (Новая группа), создайте новую группу с заголовком «Summary» (Сводка) и с именем «summary» и сохраните форму.
5. Щелкните на ссылке configure (настроить) рядом с группой Summary (Сводка) (*admin/content/node-type/review/groups/group_summary*), определите ее стиль, как collapsible (сворачиваемая), и сохраните форму.
6. Вернитесь на вкладку manage fields (управлять полями) (*admin/content/node-type/review/fields*), создайте новое поле, используя настройки из табл. 4.2.

Таблица 4.2. Настройки для поля Pros (За)

Параметр настройки	Значение
Label (Заголовок)	Pros (За)
Field name (Имя поля)	pros
Field type (Тип поля)	Text (Текст)
Widget type (Тип виджета)	Text area (Текстовая область)

7. На следующей странице, где все дополнительные настройки можно оставить со значениями по умолчанию, щелкните на кнопке Save

field settings (Сохранить настройки поля). Далее повторите эти действия, чтобы создать второе поле, используя те же значения параметров настройки, но с заголовком «Cons» (Против) и с именем «cons».

8. Мы добавили в тип содержимого группу и поля. Теперь необходимо добавить поля Pros (За) и Cons (Против) в группу Summary (Сводка) и переместить их в начало формы. Расположите поля в следующем порядке (Поля Pros (За) и Cons (Против) должны оказаться внутри группы Summary (Сводка)) и по окончании щелкните на кнопке Save (Сохранить).
   - Headline (Заголовок)
   - Menu settings (Настройки меню)
   - Summary (Сводка)
     - Pros (За)
     - Cons (Против)
   - Review (Обзор)

Теперь, когда у нас появился тип содержимого Product Reviews (Обзор продукта), необходимо добавить разрешения, чтобы дать пользователям, наделенным соответствующими полномочиями, создавать материалы этого типа. Все друзья Боба и Сары будут иметь собственные учетные записи и как редакторы сайта, будут иметь возможность отправлять и редактировать собственные обзоры. Сами же Боб и Сара будут заниматься администрированием сайта и потому будут обладать возможностью редактировать или удалять любые материалы, чтобы сохранить сайт в чистоте и порядке. Перейдите на страницу Administer (Администрирование)→User management (Управление пользователями)→Permissions (Разрешения) (*admin/user/permissions*) и выполните настройки в соответствии с табл. 4.3. Щелкните на кнопке Save permissions (Сохранить права доступа) по окончании.

Таблица 4.3. Разрешения для типа содержимого Product Review (Обзор продукта)

Разрешение	**anonymous user (анонимный пользователь)**	**authenticated user (зарегистрированный пользователь)**	**editor (редактор)**	**site administrator (администратор сайта)**
модуль node				
create review content (создавать материал типа review)			отмечен	отмечен

delete any review content (удалять любой материал типа review)				отмечен
delete own review content (удалять собственный материал типа review)			отмечен	отмечен
edit any review content (изменять любой материал типа review)				отмечен
edit own review content (изменять собственный материал типа review)			отмечен	отмечен

Закончив настройки, перейдите на страницу Create content (Создать материал)→Product Review (Обзор продукта) (*node/add/review*) и создайте простой обзор. Форма создания нового обзора должна выглядеть, как изображено на рис. 4.3.

Рис. 4.3. *Создание нового обзора продукта*

Тип содержимого Product review (Обзор продукта) получился достаточно функциональным – теперь у нас имеется возможность создавать новые обзоры, заполнять списки «за» и «против» и отображать их на главной странице сайта. В окончательном виде обзор должен выглядеть примерно так, как изображено на рис. 4.4. В следующем разделе мы добавим более полную информацию о продукте, взяв ее прямо с сайта Amazon.com.

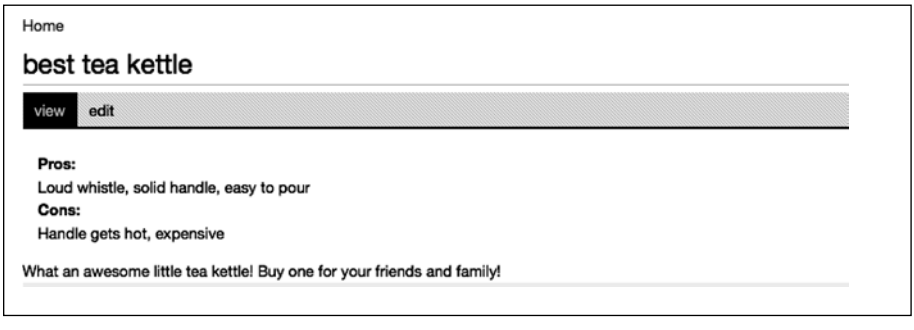


Рис. 4.4. *Законченный обзор, содержащий основную информацию*

В центре внимания: модуль Amazon

Amazon.com – одна из крупнейших интернет-компаний, предоставляющих другим сайтам открытый доступ к своим базам данных с информацией о продуктах. В случае с сайтом Super Duper Chefs нам необходимо извлекать для отображения на нашем веб-сайте такие сведения о продуктах, как фотографии, стоимость и информацию о производителе. Модуль Amazon для Drupal позволяет нам добиться этого.

Что включено?

Модуль Amazon в действительности представляет собой целую коллекцию модулей, каждый из которых предназначен для решения своего круга задач:

- Базовый модуль Amazon API служит для организации взаимодействий с веб-сайтом Amazon.com и обеспечивает своевременное обновление информации о стоимости продуктов. Все остальные модули в пакете опираются на этот модуль.
- Модуль Amazon Media сохраняет дополнительную информацию об определенных типах продуктов. Например, он отвечает за сохранение и отображение рейтинга MPAA для видеороликов и реализацию консоли для запуска видеоигр.
- Модуль Amazon Search добавляет возможность отыскивать на сайте Amazon.com продукты со страницы поиска. предоставляемой Drupal

- Модуль Amazon Field позволяет администраторам добавлять в содержимое любого типа поле, которое хранит идентификационный номер продукта на Amazon.com, и отображать фотографию продукта, извлекая ее непосредственно с сайта Amazon.com. Мы будем использовать этот модуль для расширения нашего типа содержимого Product Review (Обзор продукта).
- Модуль Amazon Filter позволяет писателям вставлять изображения продуктов и информацию о них в любом месте внутри содержимого, используя тег `[amazon]`. Это удобно для блоггеров или писателей, желающих поместить ссылку на продукт, так как не требует создавать отдельное поле для ссылки на продукт.

Модуль Amazon не требует сложной настройки, но в нем имеется несколько важных параметров, которые большинство администраторов сайтов пожелают изменить; они расположены на странице Administer (Администрирование)→Site configuration (Настройка сайта)→Amazon API (*admin/settings/amazon*) и изображены на рис. 4.5.

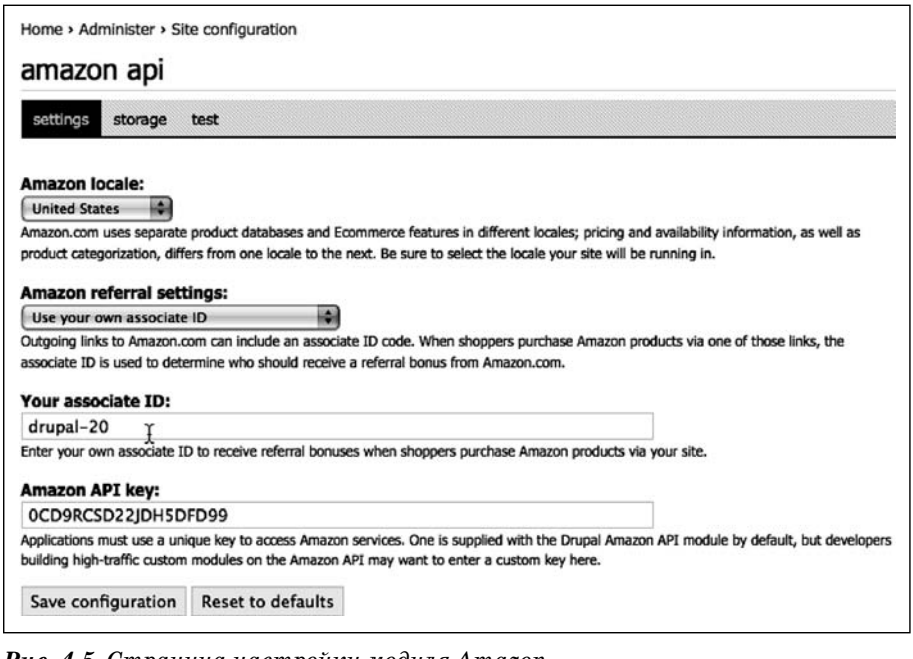


Рис. 4.5. *Страница настройки модуля Amazon*

Региональные настройки

Поскольку для каждой страны, где действует компания Amazon, имеется отдельная база данных с информацией о продуктах, ценах и со сведениями о возможности приобретения, вам наверняка потребуется на

своем веб-сайте определить значение параметра Locale (Регион). Этот параметр определяет, какой из веб-сайтов компании Amazon (*http://www.amazon.com*, *http://www.amazon.jp* и так далее) будет использоваться для поиска информации о заданном продукте.

Кроме того, всякий раз, когда на вашем сайте будут генерироваться ссылки на Amazon.com, они будут указывать на местный сайт Amazon для выбранного значения параметра Locale (Регион). Если вы не выберете другой регион, модуль Amazon будет предполагать, что ваш сайт действует в Соединенных Штатах.

Настройки ссылочной информации

Это не является обязательным, но если вы настроите параметр Amazon referral settings (Ссылочная информация для Amazon), значение Associate ID (Присвоенный идентификатор) для которого можно получить по адресу *http://affiliate-program.amazon.com/gp/associates/join*, то компания Amazon сможет выплачивать вам некоторые суммы, когда ваши посетители будут щелкать на ссылке, ведущей на сайт Amazon.com, и приобретать там какие-либо товары. Вы также можете проявить щедрость и автоматически направлять комиссионные от покупок на поддержку проекта Drupal, сообщив модулю Amazon значение параметра Drupal Association's ID (Идентификатор ассоциации Drupal).

Практика: добавление поля Amazon

В предыдущем разделе этой главы мы определили новый тип содержимого для наших обзоров продуктов. Теперь мы готовы добавить дополнительное поле, в котором будет храниться ссылка на продукт на сайте Amazon.com. Но, прежде чем добавить новое поле в наш тип содержимого, нам необходимо сделать следующее:

1. Прежде всего перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Modules (Модули) (*admin/build/modules*) и включите следующие модули:
   - Amazon
     - Amazon API
     - Amazon Field
2. Затем перейдите на страницу Administer (Администрирование)→Site configuration (Настройка сайта)→Amazon API (*admin/settings/amazon*). Выберите свой регион в списке Locale (Регион) и укажите свой идентификатор Amazon Associate ID, если он у вас имеется. Если вы находитесь в Соединенных Штатах и не имеете идентификатора Amazon Associate ID, можете оставить значения по умолчанию.

Инструменты для работы с веб-службами

Модуль Amazon – это пример программного обеспечения, которое обычно называется модулем API. Он предоставляет прикладной программный интерфейс (Application Programming Interface), дающий разработчикам Drupal доступ к данным или к функциональным возможностям других веб-сайтов. Похожие модули позволяют сайтам на базе Drupal получать карты от веб-службы Google Maps, отправлять сообщения службе микроблоггинга Twitter, синхронизировать даты и события с сайтом Upcoming.org и многое другое.

Сотни таких модулей API можно найти в категории «Third-party integration» (Сторонние компоненты), в разделе загрузки на сайте *http://drupal.org* (*http://drupal.org/project/modules/category/52*). Если вы хотите связать свой сайт на базе Drupal с каким-либо популярным веб-сайтом, вам определенно стоит ознакомится с содержимым этой страницы.

Добавление поля Product ID

Закончив настройку модуля Amazon, можно продолжить настройку типа содержимого Product Review (Обзор продукта):

1. Перейдите на страницу Administer (Администрирование)→Content management (Содержание)→Content types (Типы материалов) (*admin/content/types*), щелкните на ссылке manage fields (управлять полями) (*admin/content/node-type/review/fields*) для типа содержимого Product Review (Обзор продукта) и создайте новое поле, используя настройки в табл. 4.4.

Таблица 4.4. Настройки поля Product ID (Идентификатор продукта)

Параметр настройки	Значение
Label (Заголовок)	Amazon Product ID (Идентификатор продукта на Amazon)
Field name (Имя поля)	product_id
Field type (Тип поля)	Amazon item (Элемент Amazon)
Widget (Виджет)	Text field (Текстовое поле)

2. Щелкните на кнопке Save (Сохранить), чтобы создать поле, после чего откроется следующая страница с настройками поля, значения которых следует определить в соответствии с табл. 4.5.

Таблица 4.5. Настройки поля Product ID (Идентификатор продукта)

Параметр настройки	Значение
Product review settings (Настройки типа содержимого Product review)	
Help text (Справочный текст)	Enter the Amazon product ID of the item you're reviewing. (Введите идентификатор продукта на Amazon для описываемого элемента)
Global settings (Общие настройки)	
Required (Обязательно)	отмечено

1. Щелкните на кнопке Save field settings (Сохранить настройки поля), чтобы завершить процедуру, после чего вы вернетесь на страницу Manage fields (Управлять полями) для типа содержимого Product review (Обзор продукта).
2. На странице Manage fields (Управлять полями) (*admin/content/node-type/review/fields*) переместите новое поле Amazon product ID (Идентификатор продукта на Amazon) так, чтобы оно следовало сразу за полями Pros (За) и Cons (Против), как показано ниже:
   - Headline (Заголовок)
   - Menu settings (Настройки меню)
   - Summary (Сводка)
     - Pros (За)
     - Cons (Против)
   - Amazon product ID (Идентификатор продукта на Amazon)
   - Review (Обзор)

Перейдите на страницу Create content (Создать материал)→Product Review (Обзор продукта) (*node/add/review*) и добавьте новый обзор. На этот раз заполните не только поля Headline (Заголовок), Pros (За), Cons (Против) и Review (Обзор), но и поле идентификатора продукта на Amazon. Новый обзор должен выглядеть примерно так, как изображено на рис. 4.6.

Позже мы задействуем настройки на вкладке Display Fields (Поля) для модуля CCK, чтобы определить, какая информация должна отображаться модулем Amazon в этих узлах.

Для еще более тонкой настройки отображения информации о продуктах модулем Amazon, в частности для таких продуктов, как книги и программное обеспечение, модуль Amazon Media (коротко будет описываться в разделе «В центре внимания...») включает несколько файлов шаблонов, которые можно изменить, чтобы извлекать определен-

ные поля. О файлах шаблонов мы будем говорить в главе 11, при обсуждении проблемы визуального оформления сайта.

Теперь наши обзоры продуктов содержат поля с перечислением достоинств и недостатков продуктов и ссылку на Amazon.com. Что осталось? Нам необходимо каким-то способом сохранить оценку автора обзора для каждого описываемого продукта и дать возможность посетителям сайта выставлять собственные оценки. Для этого рассмотрим модули Voting API и Fivestar.

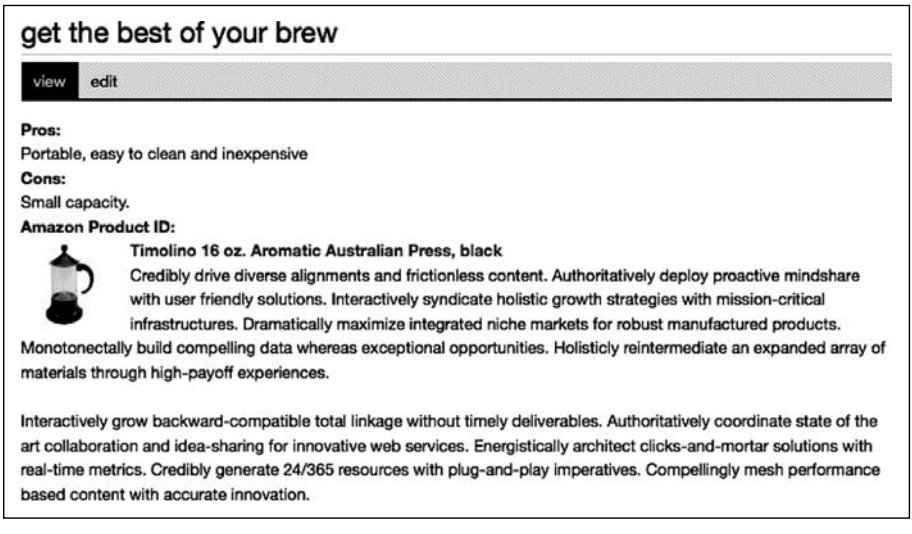


Рис. 4.6. *Обзор с информацией о продукте, полученной с сайта Amazon.com*

Поиск идентификаторов продуктов

Всякий раз после ввода идентификатора продукта наше поле связи с сайтом Amazon будет автоматически загружать фотографию продукта и информацию о цене. Это конечно хорошо, но как обозреватели на сайте будут отыскивать эти идентификаторы?

Самый простой способ состоит в том, чтобы отыскать продукт на веб-сайте Amazon.com, используя его собственную функцию поиска. На Amazon.com для каждого продукта имеется отдельная страница, а идентификатор продукта обычно можно увидеть в двух местах: в адресе URL страницы и в разделе Product details (Сведения о продукте) на странице, под заголовком ASIN (Amazon Standard Identification Number – стандартный идентификационный номер Amazon). Примеры приводятся на рис. 4.7 и рис. 4.8.

Модуль Amazon гарантирует, что любые идентификаторы продуктов, введенные в обзоры на сайте Super Duper Chefs, будут указывать на существующие описания продуктов на сайте Amazon.com.

Рис. 4.7. *Адрес URL страницы с описанием продукта на сайте Amazon.com с выделенным идентификатором*

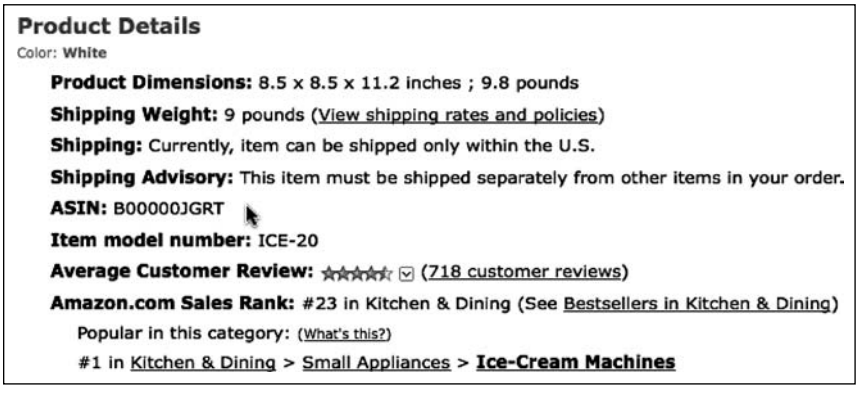


Рис. 4.8. *Раздел со сведениями о продукте на странице с описанием*

В центре внимания: Voting API и Fivestar

На крупных веб-сайтах посетителям очень часто предоставляется возможность оценивать содержимое. Кроме того что эта возможность позволяет посетителям сразу же переходить к просмотру материалов, обладающих наивысшими оценками, она также позволяет вам, как администратору сайта, определять, какие материалы на вашем сайте привлекают больше внимания.

Практически все модули для Drupal, позволяющие выставлять рейтинги или оценки, опираются на использование модуля Voting API. Хотя этот модуль сам по себе не предлагает дополнительных особенностей для вашего сайта, тем не менее, он дает в руки разработчиков набор инструментов для построения системы оценок и реализует стандартный способ хранения голосов и вычисления результатов. Это позволяет разработчикам сконцентрироваться на том, что делает их работу неповторимой (например, на представлении результатов голосования новым способом), а модуль Voting API выполнит всю черную работу.

Одним из преимуществ такой организации, когда модули опираются на использование модуля Voting API, является возможность совместного использования одних и тех же данных. Например, модули jRating (*http://drupal.org/project/jrating*), Criteria Rating (*http://drupal.org/project/criteria_rating*) и Fivestar (*http://drupal.org/project/fivestar*) предлагают немного разные возможности, но все они преследуют одну и ту же цель: оценить содержимое в определенной шкале оценок и ото-

бразить текущую среднюю оценку в стиле отображения рейтингов на сайте Amazon.com, то есть в виде последовательности звездочек. Хотя работа с этими модулями немного отличается и они дают администраторам различные варианты представления и использования результатов голосования, тем не менее, они могут использоваться совместно. Голоса, сохраненные одним модулем, могут использоваться любым другим из перечисленных, так как для сохранения и управления данными все они используют модуль Voting API.

Модуль Fivestar предлагает массу параметров настройки, от вида и цвета звездочек, используемых для отображения оценки, до того, как будут представлены результаты при просмотре посетителями нового материала. На рис. 4.9 показаны варианты виджетов отображения оценок, предоставляемых модулем Fivestar. Виджет, который посетители смогут использовать для выставления своей оценки, может отображаться в самом материале, в плавающем блоке и даже в форме создания комментария, с помощью которой посетители отправляют свои комментарии.

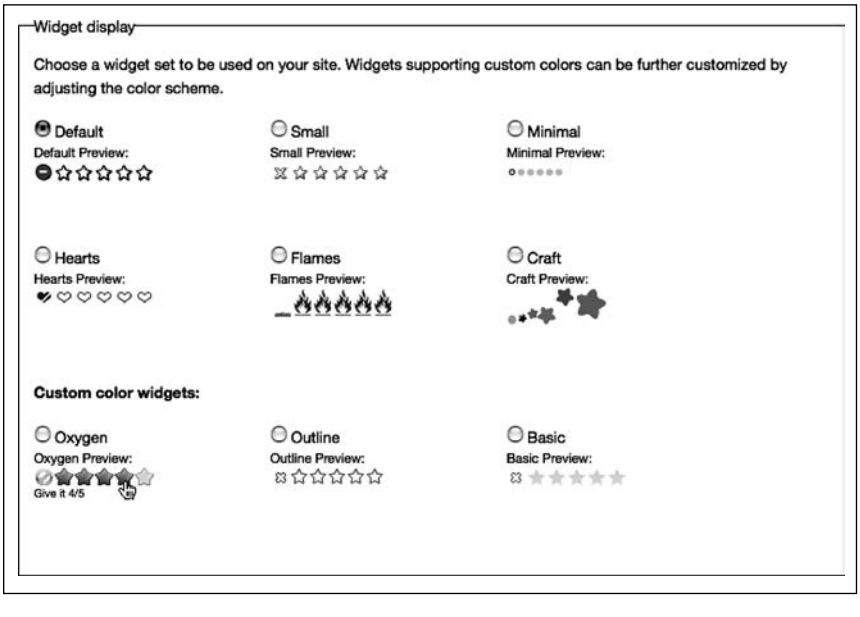


Рис. 4.9. *Выбор виджетов отображения оценки в модуле Fivestar*

Несмотря на свое название (которое переводится, как «пять звездочек»), модуль Fivestar может отображать произвольное количество звездочек: 10 звездочек, 3 звездочки и даже может использовать шкалу с одной звездочкой. Кроме того, он предоставляет собственный тип поля для модуля CCK: простое числовое поле для любого материала, которое может

отображаться с использованием собственного виджета модуля Fivestar отдельно от оценок, полученных с помощью Voting API.

Практика: добавление оценок

На сайте Super Duper Chefs мы будем использовать обе уникальные особенности модуля Fivestar: статическое поле Rating (Оценка) в типе содержимого Product Review (Обзор продукта) и виджет голосования, присоединенного к форме отправки комментария для любого обзора продукта. Такой подход позволит хранить в каждом обзоре отдельно официальную оценку и отдельно результаты голосования читателей.

Для начала перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Modules (Модули) (*admin/build/modules*) и включите следующие модули:

- Voting
  - Fivestar
  - Fivestar Comments
  - Voting API

Добавление поля Rating

Перейдите на страницу Administer (Администрирование)→Content management (Содержание)→Content types (Типы материалов) (*admin/content/types*), щелкните на ссылке manage fields (управлять полями) для типа содержимого Product Review (Обзор продукта) (*admin/content/nodetype/review/fields*) и создайте новое поле, используя настройки из табл. 4.6.

Таблица 4.6. Создание поля Rating (Оценка)

Параметр настройки	Значение
Label (Заголовок)	Rating (Оценка)
Field name (Имя поля)	rating
Field type (Тип поля)	Fivestar Rating
Widget (Виджет)	Stars

Щелкните на кнопке Save (Сохранить), чтобы создать поле, после чего откроется следующая страница с настройками поля. Установите значения параметров в соответствии с табл. 4.7 и щелкните на кнопке Save field settings (Сохранить настройки поля). Поскольку это редакционная оценка, пользователи не должны изменять это значение, поэтому мы уберем такую возможность. Мы также добавим это поле в группу

Summary (Сводка), чтобы оценка отображалась вместе с информацией «за» и «против».

Когда вы вернетесь на вкладку Manage fields (Управлять полями) (*admin/content/node-type/review/fields*), поместите новое поле Rating (Оценка) в группу Summary (Сводка), выше полей Pros (За) и Cons (Против). После того как вы щелкнете на кнопке Save (Сохранить), список должен выглядеть, как показано ниже:

- Headline (Заголовок)
- Menu settings (Настройки меню)
- Summary (Сводка)
  - Rating (Оценка)
  - Pros (За)
  - Cons (Против)
- Amazon product ID (Идентификатор продукта на Amazon)
- Review (Обзор)

Таблица 4.7. Дополнительные настройки поля Rating (Оценка)

Параметр настройки	Значение
Product review settings (Настройки типа содержимого Product review)	
Allow user to clear value (Разрешить пользователю очищать значение)	не отмечено
Global settings (Общие настройки)	
Required (Обязательно)	отмечено
Number of stars (Количество звездочек)	5

Включение оценок посетителей

Щелкните на ссылке edit (изменить) для типа Product review (Обзор продукта) (*admin/content/node-type/review*) и раскройте раздел Fivestar ratings (Оценки Fivestar) с параметрами. После того как будет отмечен флажок Enable Fivestar rating (Включить оценку Fivestar), на странице появится изображение виджета, изображающего оценку, внешний вид которого зависит от выбранных настроек.

Не бойтесь экспериментировать с различными настройками: модуль Fivestar поддерживает просмотр внешнего вида виджета, отражающего ваш выбор. Выполните настройки в соответствии со значениями в табл. 4.8, как изображено на рис. 4.10. С этими настройками пользователь будет видеть свою собственную оценку, при ее наличии, или среднюю оценку по всем пользователям.

Щелкните на кнопке Save content type (Сохранить тип содержимого), чтобы сохранить внесенные изменения.

Не забывайте, что у вас есть возможность поиграть с настройками на странице Administer (Администрирование)→Site configuration (Настройка сайта)→Fivestar (*admin/settings/fivestar*) и выбрать другие изображения, такие как сердечки или языки пламени, которые будут использоваться посетителями для оценки содержимого. Если вы выберете виджет из раздела Custom color widgets (Виджеты с настраиваемым цветом), вы даже сможете выбирать цветовую схему, соответствующую оформлению сайта.

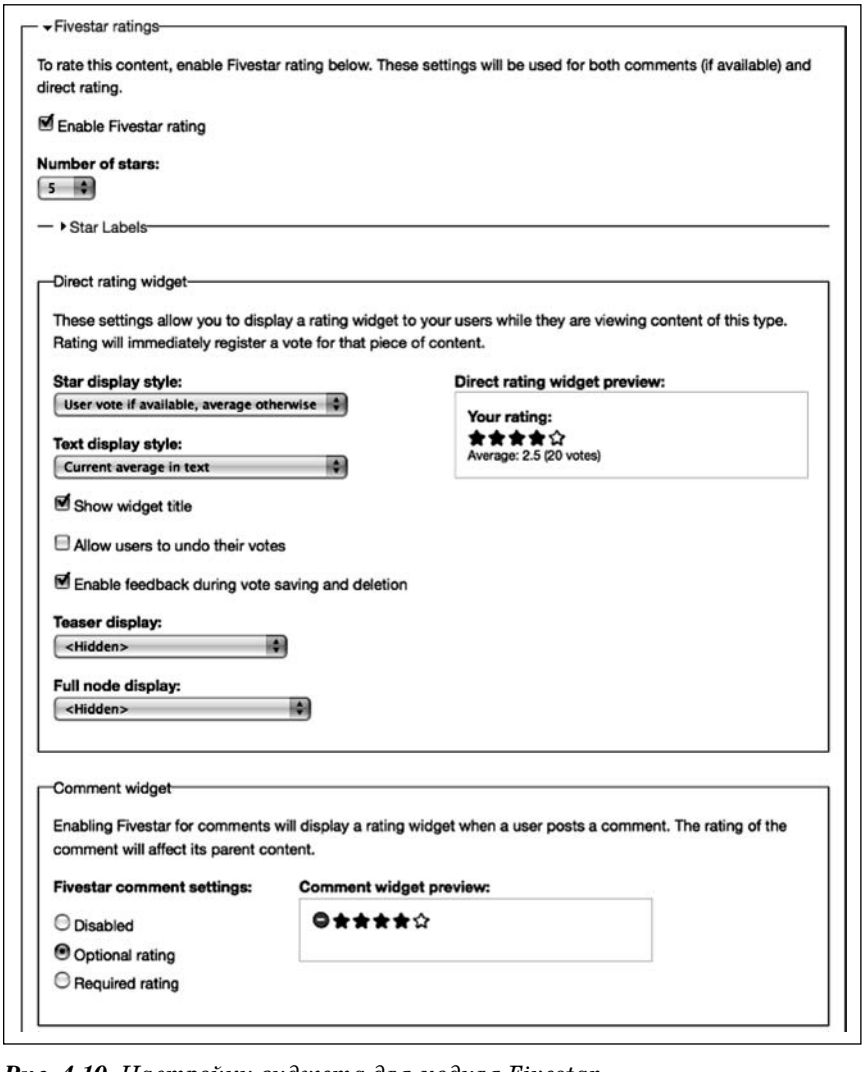


Рис. 4.10. *Настройки виджета для модуля Fivestar*

Таблица 4.8. Настройки оценки Fivestar для типа содержимого Product Review (Обзор продукта)

Поле	Значение
Direct rating widget (Виджет прямого голосования)	
Star Display Style (Стиль отображения звездочек)	User vote if available, average otherwise (Оценка пользователя, если имеется, иначе среднее значение)
Text Display Style (Стиль отображения текста)	Current average in text (Текущее среднее значение в тексте)
Full node display (Отображение всего материала)	<Hidden> (<Скрыто>)
Comment widget (Виджет комментария)	
Fivestar comment settings (Настройки комментария Fivestar)	Optional rating (Необязательная оценка)

Настроив оценки, нам необходимо дать пользователям возможность выставлять оценку. Перейдите на страницу Administer (Администрирование)→User management (Управление пользователями)→Permissions (Разрешения) (*admin/user/permissions*) и установите разрешения в соответствии с табл. 4.9. По окончании щелкните на кнопке Save permissions (Сохранить права доступа).

Таблица 4.9. Разрешения для оценок Fivestar

Разрешение	anonymous user (анонимный пользователь)	authenticated user (зарегистрированный пользователь)	editor (редактор)	site administrator (администратор сайта)
модуль fivestar				
rate content (оценивать материалы)		отмечен	отмечен	отмечен

Теперь все готово для обзоров. Обозреватели сайта теперь могут писать обзоры, включающие описания «за» и «против», оценивать продукт, используя интуитивно понятный масштаб, и вставлять полную информацию о цене и производителе с сайта Amazon.com. Кроме того, пользова-

тели могут оставлять свои комментарии о продукте и давать свои оценки. На рис. 4.11 показаны наши новые оценки Fivestar в действии.

Рис. 4.11. *Вид страницы с обзором продукта, содержащим оценку обозревателя, в процессе создания комментария пользователем*

Практика: создание представления со списком продуктов

Теперь, когда у нас имеется несколько продуктов, нам необходимо добавить страницу со списком этих продуктов, чтобы позволить посетителям просматривать все продукты, для которых имеются обзоры, сравнивать официальную оценку с оценкой посетителей и сортировать по различным критериям, как показано на рис. 4.12. Эту страницу легко можно создать с помощью модуля Views.

product finder

Title	Price	Official rating	Reader rating
Cuisinart Chef's Classic Stainless 2-Quart Saucepan with Cover	$50.00	☆☆☆☆☆	☆☆☆☆☆
Timolino 16 oz. Aromatic Australian Press, black	$17.00	☆☆☆☆☆	☆☆☆☆☆

Рис. 4.12. *Представление Product finder*

1. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Modules (Модули) (*admin/build/modules*) и включите следующие модули:
   - Other (Другой)
     – Advanced help
   - Views
     – Views
     – Views UI
2. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Views, щелкните на вкладке Add (Добавить) (*admin/build/views/add*) и заполните настройки представления значениями из табл. 4.10.

Таблица 4.10. Параметры создания представления Products

Параметры настройки представления	**Значение**
View name (Название представления)	products
View description (Описание представления)	List of reviewed products (Список продуктов, для которых имеются обзоры)
View tag (Рубрика представления)	products
View type (Тип представления)	Node (Материал)

3. После щелчка на кнопке Next (Далее) определите значения параметров представления в разделе Basic settings (Базовые настройки) в соответствии с табл. 4.11.

Таблица 4.11. Базовые настройки представления Product finder (Список продуктов)

Defaults: Basic settings (По умолчанию: Базовые настройки)	**Значение**
Title (Название)	Product finder (Список продуктов)
Style (Стиль)	Table (Таблица)

4. После определения значений в разделе Basic settings (Базовые настройки) вы увидите сообщение: «You need at least one field before you can configure your table settings» («Вам необходимо создать по меньшей мере одно поле, прежде чем определять настройки таблицы»). Мы сделаем это чуть позже, но сначала нам нужно создать отношение для наших полей, чтобы мы могли задействовать в представлении поля модуля Amazon и отобразить среднюю оценку для каждого продукта.

   Щелкните на ярлыке с изображением знака плюс в разделе Relationships (Отношения) и отметьте отношения Content: Amazon item: Amazon product ID (field_product_id) (Содержимое элемента Amazon: идентификатор продукта на Amazon (field_product_id)) и Node: Voting results (Узел: Результаты голосования). Щелкните на кнопке Add (Добавить), чтобы добавить отношение, и заполните параметры настройки в соответствии с табл. 4.12. Не забывайте щелкать на кнопке Update (Обновить) после настройки каждого поля.

Таблица 4.12. Настройки отношения Voting Results (Результаты голосования) для представления Product finder

Defaults: Relationships (По умолчанию: Отношения)	**Значение**
Content: Amazon item: Amazon product ID (field_product_id) (Содержимое элемента Amazon: идентификатор продукта на Amazon (field_product_id))	(Изменения не требуются)
Node: Voting results (Узел: Результаты голосования)	Value type (Тип значения): Percent (Проценты) Vote tag (Тег оценки): Vote (Оценка) Aggregate function (Агрегатная функция): Average (Среднее)

5. В разделе Fields (Поля) в интерфейсе модуля View щелкните на ярлыке с изображением знака плюс, отметьте следующие поля и щелкните на кнопке Add (Добавить). В результате появятся поля с названием продукта, ценой, официальной оценкой и оценкой пользователей:
   - Amazon: List price (Amazon: цена)
   - Amazon: Title (Amazon: Название)
   - Content: Fivestar Rating: Rating (field_rating) (Содержимое: Оценка Fivestar: Оценка (field_rating))
   - Voting API results: Value (Результаты Voting API: Значение)
6. Определите настройки для каждого нового поля в соответствии с табл. 4.13. Не забывайте щелкать на кнопке Update (Обновить) после настройки каждого поля.

Таблица 4.13. Настройки полей для представления Product finder (Список продуктов)

Defaults: Field configure settings (По умолчанию: Параметры настройки полей)	**Значение**
Amazon: List price (Amazon: цена)	Label (Название): Price (Цена)
Amazon: Title (Amazon: Название)	Link behavior (Поведение ссылки): A link to the node the product is associated with (Ссылка на узел, ассоциированный с продуктом)
Content: Fivestar Rating: Rating (field_rating) (Содержимое: Оценка Fivestar: Оценка (field_rating))	Label (Название): Custom (Собственное)Custom label (Собственное название): Official rating (Официальная оценка)
Voting API results: Value (Результаты Voting API: Значение)	Appearance (Внешний вид): Fivestar Stars (display only) (Звездочки Fivestars (только для отображения)) Relationship (Отношение): Voting results (Результаты голосования)Label (Название): Reader rating (Оценка посетителей)

7. Давайте определим другой порядок следования полей. Щелкните на ярлыке ↑↓ (обозначающем операцию переупорядочения) в разделе Fields (Поля). Переместите поле (field_product_id_asin) Amazon: Title ((field_product_id_asin) Amazon: Название) в начало списка, чтобы название продукта следовало первым.
8. Теперь, закончив настройку полей, щелкните на ярлыке с изображением шестеренки рядом с параметром Style (Стиль), чтобы настроить таблицу, и установите значения настроек для таблицы с поддержкой функции сортировки в соответствии с табл. 4.14. По окончании щелкните на кнопке Update (Обновить).

Таблица 4.14. Параметры настройки стиля таблицы для представления Product finder (Список продуктов)

Defaults: Table style options (По умолчанию: Настройки стиля таблицы)	**Значение**
Sortable (Сортировка)	Все флажки отмечены
Default sort (Сортировка по умолчанию)	Official rating (Официальная оценка)

9. В разделе Filters (Фильтры) щелкните на ярлыке с изображением знака плюс и отметьте фильтры Node: Published or admin (Узел: Опубликовано или в административном разделе) и Node: Type (Узел: Тип). С такими настройками отображаться будут только опубликованные обзоры, если, конечно, вы не зарегистрируетесь как пользователь, обладающий возможностью просматривать неопубликованные материалы (с разрешением administer nodes (управлять материалами)). Щелкнув на кнопке Add (Добавить), настройте новые поля в соответствии с табл. 4.15, не забывая щелкать на кнопке Update (Обновить) после настройки каждого поля.

Таблица 4.15. Настройки фильтров для представления Product finder (Список продуктов)

Defaults: Filter setting (По умолчанию: Настройки фильтров)	**Значение**
Node: Published or admin (Узел: Опубликовано или в административном разделе)	Не требует настройки
Node: Type (Узел: Тип)	Node type (Тип материала): Product review (Обзор продукта)

10. Наконец, используя ссылки, расположенные слева на странице с настройками модуля Views, добавьте в представление новое отображение Page (Страница). Установите значения настроек отображения Page (Страница) в соответствии с табл. 4.16.

Таблица 4.16. Настройки отображения Page (Страница) для представления Product finder (Список продуктов)

Page: Page settings (Страница: Настройки отображения)	**Значение**
Path (Путь)	products
Menu	
Type (Тип)	Normal menu entry (Обычный элемент меню)
Title (Заголовок)	Product finder (Список продуктов)

По окончании выполнения всех этих действий сохраните представление, которое теперь должно выглядеть, как изображено на рис. 4.13.

С настройками, которые мы только что выполнили, должна появиться ссылка Product finder (Список продуктов) в навигационном меню сайта. После щелчка на этой ссылке должен появиться аккуратный список всех обзоров, имеющихся на сайте, в сопровождении выведенных

рядом официальной и пользовательской оценок, как изображено на рис. 4.12 выше.

В нашем списке необходимых шагов остался невыполненным единственный пункт: реализация функции поиска по списку продуктов, чтобы посетители могли отыскивать интересующие их товары.

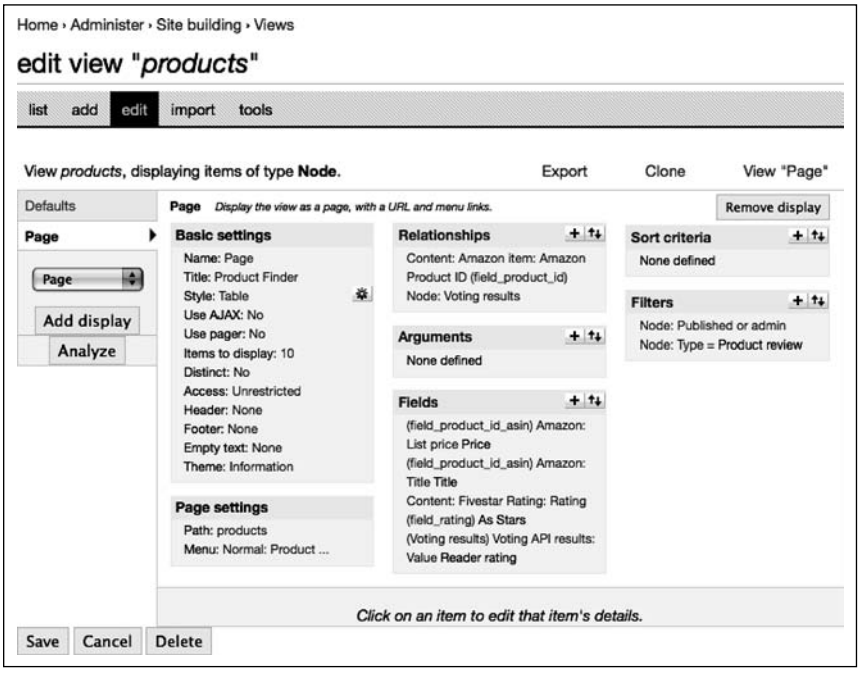


Рис. 4.13. *Окончательные настройки представления Product Finder (Список продуктов)*

В центре внимания: модуль Search

Модуль *Search*, встроенный в систему Drupal, предлагает мощные, гибкие возможности поиска и ранжирования результатов. За кулисами этот модуль строит индексы всех слов, встречающихся в содержимом сайта. Когда пользователь выполняет поиск по фразе, содержимое ранжируется с применением настраиваемых правил и отображается в порядке релевантности. На любом сайте, работающем под управлением Drupal, имеется возможность подкорректировать эти правила, перейдя на страницу Administer (Администрирование)→Site Configuration (Настройка сайта)→Search Settings (Настройка поиска) (*admin/settings/search*) и изменив веса в разделе Content Ranking (Ранжирование материалов), как показано на рис. 4.14.

Модуль Search также предлагает дополнительные возможности для сайтов с большим числом материалов. На странице Advanced Search (Рас-

ширенный поиск), изображенной на рис. 4.15, пользователям предоставляется возможность уточнять искомое содержимое, выбирая типы материалов, в которых следует производить поиск, указывая ключевые слова и определяя другие критерии.

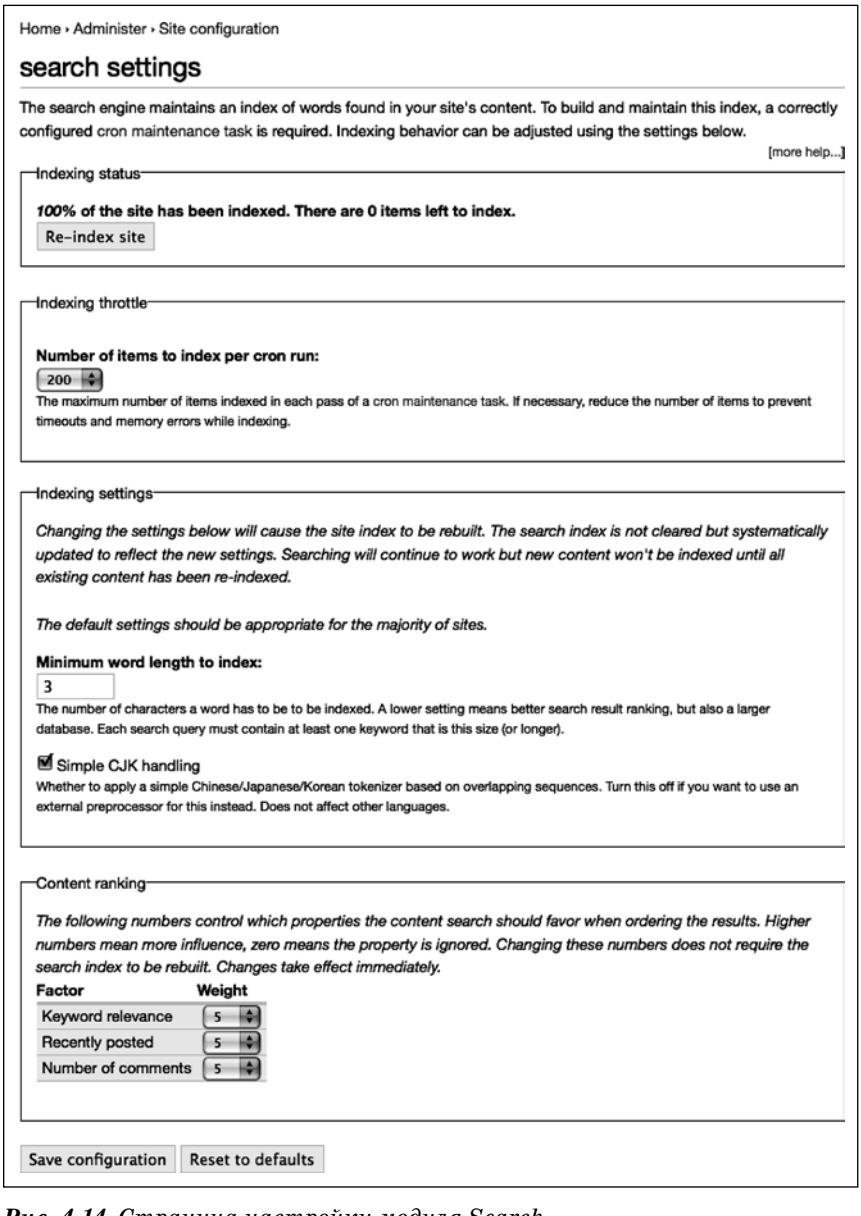


Рис. 4.14. *Страница настройки модуля Search*

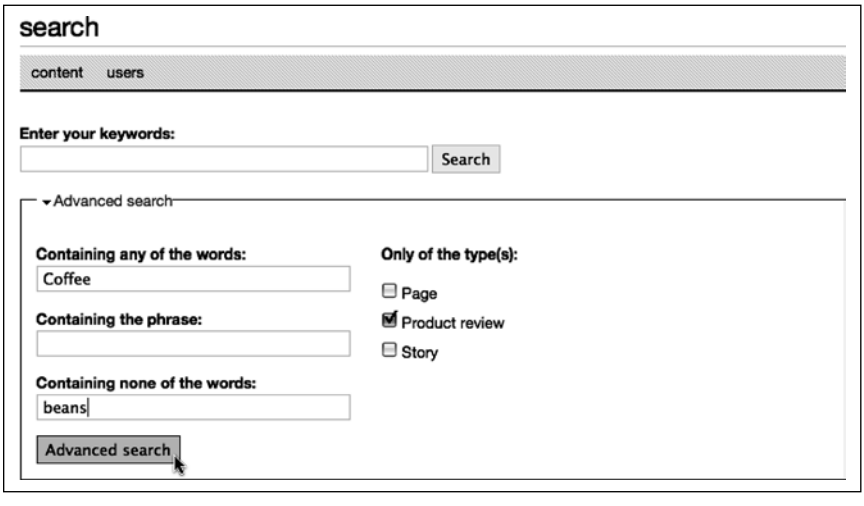


Рис. 4.15. *Страница Advanced Search (Расширенный поиск) в действии*

Важность выполнения регулярных процедур

Процедура индексирования, используемая модулем Search, работает только при корректной настройке утилиты *cron*. Это утилита, запускающая различные команды на вашем веб-сервере через заданные интервалы времени. Она используется для выполнения таких действий по обслуживанию сайта, как удаление старых записей из файлов журналов, массовая рассылка по электронной почте и других, которые необходимо выполнять регулярно.

Каждый раз, когда запускается утилита cron, система Drupal индексирует некоторую часть содержимого сайта – по умолчанию 200 сообщений каждый раз. Если на вашем сайте уже имеется значительное число сообщений, скорость индексирования будет зависеть от того, с какой частотой настроен запуск процедуры индексирования в утилите `cron`.

Вместо утилиты cron можно установить модуль Poormanscron (*http://drupal.org/project/poormanscron*). Этот модуль проверяет необходимость выполнения запланированных действий незаметно для посетителей вашего веб-сайта. Каждый раз, когда на веб-сайт приходит очередной посетитель, модуль Poormanscron проверяет, назрела ли необходимость выполнить какие-либо действия с момента последнего его запуска, и если такая необходимость имеется, – выполняет предусмотренные операции. Эти проверки выполняются после того, как страница будет загружена, поэтому посетитель не увидит никакой разницы.

Конечно, такой прием будет работать, только если на сайт регулярно приходят посетители. Но, с другой стороны, если посетители приходят редко, не имеет значения, как часто будет обновляться поисковый индекс.

Если вы не уверены, была ли настроена утилита cron, или вы ведете разработку и тестирование сайта на локальном компьютере, вы можете предложить системе Drupal выполнить эти задачи, перейдя на страницу Administer (Администрирование)→Reports (Отчеты)→Status report (Отчет о состоянии) (*admin/logs/status*) и щелкнув на ссылке run cron manually (выполнить регулярные процедуры вручную). За дополнительной информацией о настройке cron для вашего сайта обращайтесь по адресу *http://Drupal.org/cron*.

Поиск с помощью модуля Views

Несмотря на то, что форма расширенного поиска дает пользователям совсем немного возможностей, тем не менее, нам очень трудно предположить, как будет эта страница выглядеть и как на ней будут отображаться результаты. Кроме того, она может представлять ужасающее количество параметров, особенно когда на сайте имеется большое число рубрик таксономии.

Модуль Views – это один из способов обеспечить более полный контроль над функцией поиска: его фильтры могут сократить перечень материалов, опираясь на механизм индексирования слов в системе поиска. Представление может, к примеру, перечислить только записи в блоге, где упоминается слово «котята». Мы будем использовать этот модуль на сайте Super Duper Chefs, чтобы добавить возможность фильтрации на страницу Product finder (Список продуктов).

Практика: добавление возможности поиска в список продуктов

Чтобы преобразовать страницу Product finder (Список продуктов) в предметный указатель с функцией поиска, мы добавим в представление два новых фильтра: один будет использоваться для фильтрации элементов списка по названию производителя, а второй будет ограничивать результаты поиска обзорами, в которых встречаются определенные слова.

Как правило, доступ к фильтрам на сайте блокируется, и они могут изменяться только администратором сайта. Однако нам необходимо дать пользователям возможность вводить свои критерии поиска. К счастью, модуль Views позволяет «выставлять» любые свои фильтры. В этом случае к заголовку отображения представления добавляется небольшая форма. Эта форма превращает представление в простой инструмент поиска, с помощью которого посетители сайта могут управлять отображением результатов с помощью фильтров модуля Views, как показано на рис. 4.16.

Ниже описаны шаги, которые необходимо выполнить, чтобы добавить в список функцию поиска:

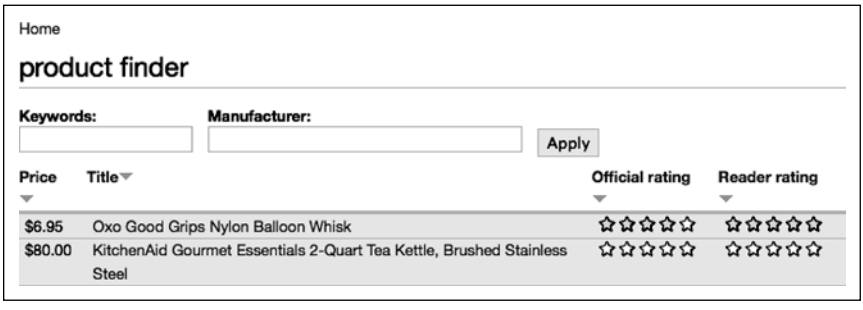


Рис. 4.16. *Представление Product finder (Список продуктов) с функцией поиска*

1. Включите поддержку функции поиска на сайте: перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Modules (Модули) (*admin/build/modules*) и включите модуль Core: Search (Ядро: Search).
2. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Views (*admin/build/views*) и щелкните на ссылке edit (изменить) для представления Product (Список продуктов).
3. Щелкните на ярлыке с изображением знака плюс в разделе Filters (Фильтры), чтобы включить фильтры Amazon: Manufacturer и Search: Search Terms.
4. В настройках каждого фильтра щелкните на кнопке Expose (Показать), что приведет к отображению фильтра в виде формы с полями, с которыми посетитель сможет взаимодействовать. Для каждого фильтра настройте параметры в группе Exposed Filter (Отображаемый фильтр) в соответствии с табл. 4.17. А для перемещения между формами с настройками, как обычно, щелкайте на кнопке Update (Обновить).

Таблица 4.17. Настройки фильтра Search Terms

Defaults: Configure filter setting (По умолчанию: Настройки фильтров)	**Значение**
Amazon: Manufacturer (Amazon: Производитель)	Operator (Оператор): Contains (Содержит) Case sensitive (Чувствительность к регистру): не отмечено Label (Название): Manufacturer: (Производитель:)
Search: Search Terms (Search: Искомые слова)	On Empty Input (При отсутствии ввода): Show All (Показать все Label (Название): Keywords: (Ключевые слова:)

5. Щелкните на ярлыке ↑↓ (обозначающем операцию переупорядочения) в разделе Filters (Фильтры) и поместите фильтр Search: Search Terms (Search: Искомые слова) выше фильтра Amazon: Manufacturer (Amazon: Производитель), чтобы поле ввода для этого фильтра отображалось на первом месте.
6. Сохраните представление, настройки которого теперь должны выглядеть, как показано на рис. 4.17.

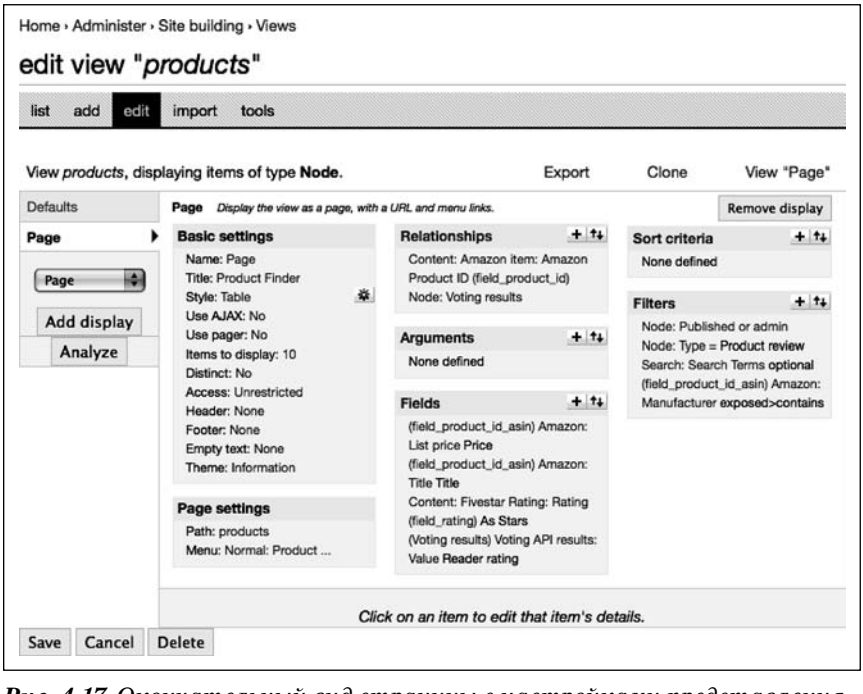


Рис. 4.17. *Окончательный вид страницы с настройками представления Product finder (Список продуктов)*

Прежде чем приступить к тестированию новой функции поиска, необходимо дать пользователям разрешения на ее использование. Нам необходимо, чтобы все посетители сайта имели возможность производить поиск. Перейдите на страницу Administer (Администрирование)→User management (Управление пользователями)→Permissions (Разрешения) (*admin/user/permissions*) и установите разрешения в соответствии с табл. 4.18. Это даст возможность пользоваться поиском всем посетителям, как зарегистрированным, так и анонимным. Сохраните разрешения.

Последнее, что нам осталось сделать, – выполнить индексирование материалов сайта, чтобы к моменту поиска ключевые слова уже были доступны. Вам определенно следует настроить для этого автоматическое выполнение регулярных процедур, тем не менее, мы выполним индексирование вручную, чтобы можно было убедиться в корректной рабо-

те нашей функции поиска. Перейдите на страницу Administer (Администрирование)→Reports (Отчеты)→Status report (Отчет о состоянии) (*admin/reports/status*), найдите в таблице строку Cron maintenance tasks (Выполнение регулярных процедур) и щелкните на ссылке run cron manually (выполнить регулярные процедуры вручную).

Таблица 4.18. Разрешения на доступ к функции поиска

Разрешение	**anonymous user (анонимный пользователь)**	**authenticated user (зарегистрированный пользователь)**	**editor (редактор)**	**site administrator (администратор сайта)**
модуль search				
search content (поиск материалов)	отмечен	отмечен		

Теперь откройте страницу представления Product finder (Список продуктов) (*http://example.com/products*). Вы должны увидеть обычную страницу с полным списком продуктов, но на этот раз с полями фильтров над списком. Введите фразу, которая встречается в ваших обзорах, и щелкните на кнопке Apply (Применить). Вы должны получить список материалов, содержащих искомую фразу, в формате, представленном на рис. 4.16 выше.

В центре внимания: модуль CSS Injector

Поддержка тем оформления в Drupal, о которой будет рассказываться в главе 11, дает дизайнерам полный контроль над отображением содержимого сайта в веб-броузерах, а собственные темы (такие как тема Nitobe, использованная при создании сайта Super Duper Chefs) помогут придать любому сайту неповторимый вид. Но иногда бывает желательно внести лишь незначительные изменения во внешний вид сайта, не используя ничего, кроме правил CSS. С помощью этих правил дизайнеры могут настраивать размер шрифта, цвет и другие характеристики отображения, не изменяя саму разметку HTML, которая определяет структуру сайта.

Модуль CSS Injector (*http://drupal.org/project/css_injector*) дает администраторам возможность добавлять такие фрагменты в административном разделе сайта, не изменяя файлы текущей темы. Такая возможность может быть полезна, например, когда на вашем сайте используется тема оформления, куда вы добавляли свои собственные правила CSS, и вы легко можете потерять эти изменения при установке новой версии темы. Сохранив правила в модуле CSS Injector, можно избежать такой неприятности даже после замены одной темы на другую.

Модуль CSS Injector предлагает ряд расширенных возможностей, включая возможность добавления правил CSS по условию, для определенных страниц. Например, если ваши правила CSS должны применяться только к главной странице, вы сможете гарантировать, что они не будут применяться ко всему сайту. Имеется также возможность указать тип носителя для ваших правил CSS, что делает возможным добавлять информацию о стиле, которая будет применяться только при выводе страницы на печать. Наконец, для каждого правила можно включить флажок Preprocess CSS (Предварительная обработка CSS), чтобы определить, должна ли система Drupal объединять эти правила с правилами CSS текущей темы. В большинстве случаев это позволяет сэкономить время, потому что веб-броузер посетителя загружает все таблицы стилей за одно обращение к веб-серверу. Если вы добавляете огромное количество кода CSS, который применяется только к одной-двум страницам, то, возможно, будет более эффективным сохранить этот код отдельно, выключив предварительную обработку. В противном случае оставьте ее включенной.

Если вы не сильны в создании каскадных таблиц стилей CSS, по адресу *http://w3schools.com/css/default.asp* вы найдете интерактивные рабочие примеры для изучения всех типов веб-технологий, в том числе и CSS. Этот замечательный ресурс достоин того, чтобы сохранить его в своих закладках.

Практика: улучшение внешнего вида

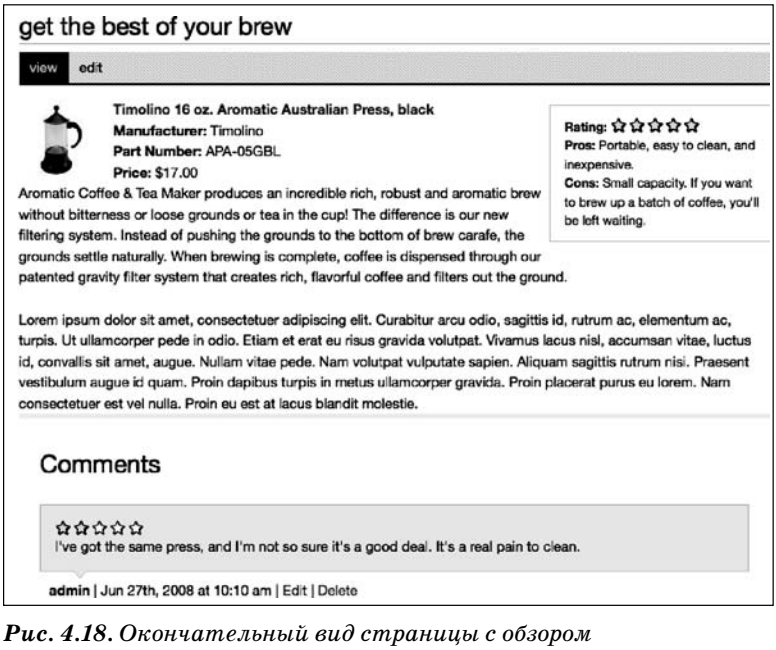


Рис. 4.18. *Окончательный вид страницы с обзором*

В этом разделе мы выполним некоторые заключительные настройки, чтобы улучшить внешний вид страницы отображения обзора, как показано на рис. 4.18.

Настройка параметров отображения полей CCK

Несмотря на то, что наш тип содержимого Product Reviews (Обзор продукта) имеет все данные и наша страница со списком продуктов выглядит просто замечательно, тем не менее, сами обзоры выглядят недостаточно привлекательно. К счастью, мы можем использовать настройки отображения модуля CCK, чтобы настроить отображение каждого поля в обзоре.

Перейдите на страницу Administer (Администрирование)→Content management (Содержание)→Content types (Типы материалов) (*admin/content/types*), щелкните на ссылке edit (изменить) для типа Product review (Обзор продукта) и затем щелкните на вкладке Display fields (Поля) (*admin/content/types/review/display*). Определите настройки полей в соответствии с табл. 4.19, что упростит отображение информации, содержащейся в обзоре. По окончании щелкните на кнопке Save (Сохранить).

Таблица 4.19. Настройки отображения для типа содержимого Product Review (Обзор продукта)

Поле	**Label (Заголовок)**	**Teaser (анонс)**	**Full (весь материал)**
Summary (Сводка)	<Hidden> (<Скрыто>)	Simple (Простой)	Simple (Простой)
Rating (Оценка)	Inline (В одной строке)	As stars (Как звездочки)	As stars (Как звездочки)
Pros (За)	Inline (В одной строке)	Default (По умолчанию)	Default (По умолчанию)
Cons (Против)	Inline (В одной строке)	Default (По умолчанию)	Default (По умолчанию)
Amazon Product ID (Идентификатор продукта на Amazon)	<Hidden> (<Скрыто>)	Small image and full info (Маленькое изображение и полная информация)	Small image and full info (Маленькое изображение и полная информация)

Настройка модуля CSS Injector

Изменение параметров отображения модуля CCK позволило немного улучшить отображение наших обзоров, но поля в группе Summary

(Сводка) все еще выглядят недостаточно привлекательно. В главе 11 мы рассмотрим некоторые основы оформления разметки HTML, которая выводится системой Drupal, а пока можем немного улучшить внешний вид, добавив несколько правил CSS с помощью модуля CSS Injector. С его помощью мы уменьшим ширину области со сводной информацией и поместим ее не в заголовок обзора, а сбоку, заключив в плавающий блок. Вот как это делается:

1. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Modules (Модули) (*admin/build/modules*) и включите модуль Other (Другой): CSS Injector.
2. Перейдите на страницу Administer (Администрирование)→Site configuration (Настройка сайта)→CSS Injector (*admin/settings/css_injector*) и щелкните на ссылке Create a new rule (Создать новое правило), чтобы добавить новое правило CSS.
3. Дайте правилу имя Floating Amazon fields (Плавающая область с полями модуля Amazon) и введите следующий текст в поле CSS code (Код CSS). Заполненная форма приводится на рис. 4.19.

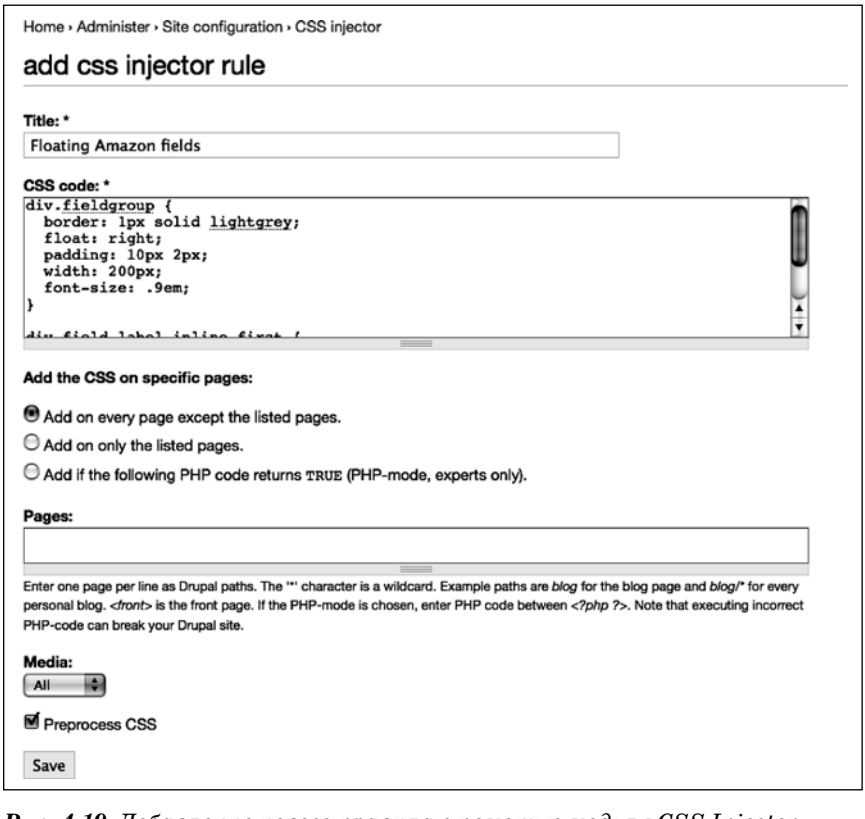


Рис. 4.19. *Добавление нового правила с помощью модуля CSS Injector*

```
div.fieldgroup {
 border: 1px solid lightgrey;
 float: right;
 padding: 10px 2px;
 width: 200px;
 font-size: .9em;
}
div.field-label-inline-first {
 float: left;
}
```

Не хочется вводить код вручную? Не беспокойтесь. Этот код также присутствует в файле *assets/ch04-reviews/amazon.css* среди примеров, распространяемых с книгой, содержимое которого легко можно скопировать и вставить.

4. Щелкните на кнопке Save (Сохранить), чтобы добавить новое правило.

В зависимости от настройки вашего веб-броузера вам может потребоваться очистить кэш броузера, чтобы увидеть изменения в таблицах стилей. Сделав это и вернувшись к одному из обзоров продуктов, добавленных ранее, можно будет убедиться, что он стал выглядеть намного привлекательнее, как изображено на рис. 4.18 выше. Какие разительные изменения за счет небольшого фрагмента кода CSS!

Дополнительно

Поздравляем! Мы закончили реализацию всех основных особенностей сайта. Если вам интересно продолжить эксперименты, можно попробовать внести некоторые улучшения с помощью других модулей Drupal.

Notifications (http://drupal.org/project/notifications)

Этот модуль дает посетителям сайта возможность подписаться на получение по электронной почте извещений о появлении новых обзоров. Он также позволяет обозревателям получать извещения о появлении новых комментариев, оставленных посетителями к их обзорам.

AdSense (http://drupal.org/project/adsense)

Этот модуль позволит Бобу и Саре размещать рекламу в боковой колонке, чтобы возместить затраты на хостинг сайта.

Blog (базовый)

Этот модуль позволит обеспечить писателей обзоров собственными блогами, в которых они смогут давать свои советы, обсуждать последние рецепты и рассматривать другие вопросы кулинарии, даже когда они не занимаются созданием обзоров.

В заключение

Что мы получили в результате проделанной работы? Мы реализовали все основные функциональные возможности, которые были затребованы Бобом и Сарой. Благодаря использованию модулей CCK, Amazon и Fivestar обозреватели могут создавать для сайта привлекательные обзоры кухонной техники. С помощью модулей Fivestar и Voting API посетители могут высказывать свое собственное мнение об этих продуктах и участвовать в рассмотрении. А благодаря интеграции модулей Views и Search они легко могут отыскивать интересующую их информацию о продуктах. Наконец, модуль CSS Injector позволил нам внести последние штрихи в оформление сайта, что придало ему блеска.

Ниже приводится список модулей, использованных в этой главе:

- AdSense: *http://drupal.org/project/adsense*
- Amazon: *http://drupal.org/project/amazon*
- CCK: *http://drupal.org/project/cck*
- CSS Injector: *http://drupal.org/project/css_injector*
- Fivestar: *http://drupal.org/project/fivestar*
- Poormanscron: *http://drupal.org/project/poormanscron*
- Views: *http://drupal.org/project/views*
- Voting API: *http://drupal.org/project/votingapi*

Ниже приводится список ресурсов, упоминавшихся в этой главе:

- Партнерская программа компании Amazon: *http://affiliate-program.amazon.com/*
- Настройка выполнения регулярных процедур: *http://drupal.org/cron*
- Модули выставления оценок: *http://drupal.org/project/Modules/category/60*
- Сторонние модули интеграции: *http://drupal.org/project/modules/category/52*
- Voting Systems Drupal group (Раздел систем голосования на сайте Drupal): *http://groups.drupal.org/voting-systems*
- Учебное руководство по CSS на сайте W3Schools: *http://w3schools.com/css/default.asp*

5

Wiki

Коллективная природа гипертекстовой среды Wiki дала начало новым интересным методам управления знаниями в сети. Википедия, самый известный пример wiki, – это интернет-энциклопедия, содержащая миллионы статей, постоянно создающихся и дополняемых тысячами добровольцев по всему миру. Посредством объединения нескольких дополнительных модулей Drupal также может использовать эти мощные особенности и объединять их с другими замечательными возможностями, которые уже присутствуют в Drupal.

Но что такое wiki? Wiki – это коллекция веб-страниц, которые могут создаваться посетителями, добавляющими и создающими содержимое с помощью своих броузеров. Среда wiki отслеживает все изменения и обладает возможностью производить откат к предыдущим версиям в случае необходимости. Для создания страниц wiki часто используется упрощенная разметка текста для обеспечения простого и быстрого форматирования. Ключевой технической особенностью является возможность создавать ссылки на страницы, которых еще не существует, что обеспечивает органичное развитие структуры сайта. Все это означает, что wiki является отличным инструментом для коллективной работы над документами. Несколько человек могут совместно работать над «оперативной» версией документа, изменяя и дополняя его по мере необходимости.

В этой главе с помощью Drupal мы создадим wiki и рассмотрим некоторые по-настоящему потрясающие возможности Drupal, касающиеся не только wiki.

В этой главе будут представлены следующие модули:

Freelinking (http://drupal.org/project/freelinking)

Обеспечивает возможность легко и просто создавать ссылки между страницами.

Markdown filter (http://drupal.org/project/markdown)

Обеспечивает простой и понятный способ ввода форматированного текста.

Diff (http://drupal.org/project/diff)

Отображает список изменений между двумя редакциями содержимого, используя выделение цветом.

Pathauto (http://drupal.org/project/pathauto)

Автоматически создает удобочитаемые и дружественные по отношению к поисковым системам адреса URL.

Token (http://drupal.org/project/token)

Вспомогательный модуль, позволяющий использовать переменные, которые позднее замещаются динамическим содержимым, такие как `[user-name]`.

Если вы собираетесь следовать за практическими упражнениями этой главы, вам следует установить Drupal, используя профиль установки Wiki из примеров кода, распространяемых вместе с книгой. В результате применения этого профиля на вашем веб-сервере будет создана заготовка веб-сайта. В окончательном виде этот веб-сайт будет выглядеть, как показано на рис. 5.1 и по адресу: *http://wiki.usingdrupal.com*. За дополнительной информацией об использовании профилей установки обращайтесь к разделу «Вступление».

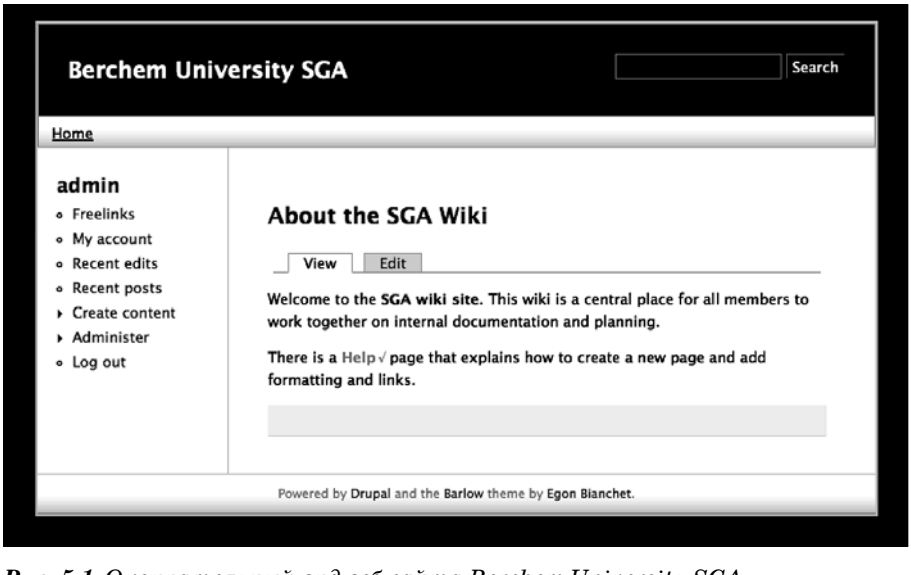


Рис. 5.1. *Окончательный вид веб-сайта Berchem University SGA*

Задача

Студенческой ассоциации органов самоуправления в университете города Берхем (Berchem University Student Government Association, SGA) требуется внутренний сайт, который мог бы обеспечить для всех членов ассоциации возможность совместной работы, доступный только членам SGA. Поскольку все они являются студентами, занятыми в различных программах, им очень трудно выкроить время для личных встреч и совместной работы. Им требуется место, где они смогли бы добавлять и дополнять информацию о внутренней политике и процедурах, извещать о встречах и планировать мероприятия. Ассоциация состоит из множества групп по всему университетскому городку, члены которых хотели бы, чтобы каждый имел возможность использовать сайт для совместной работы, но при этом имеется некоторый объем специфической информации, которая должна быть доступна для изменения только отдельным должностным лицам.

Большинство членов ассоциации не знакомы с языком разметки HTML, однако им необходимо иметь возможность форматировать текст своих документов и создавать ссылки как на существующие, так и на пока отсутствующие страницы.

Им также необходима возможность просматривать изменения, сделанные с течением времени, и возвращаться обратно к предыдущим редакциям. Они также хотели бы, чтобы страницам присваивались адреса URL, состоящие из простых для запоминания слов вместо числовых значений (например, *http://example.com/annual-blood-drive* вместо *http://example.com/node/123*).

Наконец, им требуется способ следить за активностью wiki, то есть возможность получать два списка страниц, один из которых содержал бы вновь добавленные страницы, а другой – страницы, недавно отредактированные.

Примечания к реализации

В следующих подразделах мы обсудим используемые модули и их реализацию.

Функциональные возможности Wiki

Существует достаточно много модулей, с помощью которых мы могли бы удовлетворить требования, предъявляемые к этому сайту. При поиске решений реализации wiki естественно будет начать с поиска модулей wiki. Существуют три модуля, содержащие слово «wiki» в своих названиях, Wiki (*http://drupal.org/project/wiki*), Liquid Wiki (*http://drupal.org/project/liquid*) и Wikitools (*http://drupal.org/project/wikitools*). Модуль

Wiki разрабатывался для объединения различных разработок, связанных с wiki, в единый пакет, но был заброшен в 2006 году и потому не представляет для нас интереса. Модули Liquid Wiki и Wikitools не являются «полными» реализациями wiki. Они оба реализуют некоторые общие потребности wiki, но для достижения большего объема функциональных возможностей опираются на другие модули. К сожалению, на момент написания этих строк модуль Liquid Wiki, похоже, также перестал поддерживаться, так как начиная с февраля 2007 года не было ни одного официального выпуска и нет никаких сведений о продолжении разработки. В результате у нас остается единственный жизнеспособный модуль – Wikitools.

К моменту написания этих строк модуль Wikitools продолжал разрабатываться и был пригоден к промышленной эксплуатации. Он обладает такими замечательными особенностями, как автоматическое перенаправление ссылок при изменении пути к материалам wiki и защита от перемещения страниц (обычные пользователи лишены возможности менять названия страниц wiki, что могло бы привести к изменению пути страницы). Все это, конечно, хорошо, но в нашем конкретном случае эти возможности не нужны. Ознакомьтесь с изменениями в модуле – как и в случае с любыми другими инструментами в мире Drupal, возможности программного обеспечения могут быстро расширяться при наличии достаточного интереса к ним.

Однако можно создать wiki и без всех этих модулей, что мы и сделаем в этой главе. Существует множество модулей, которые обладают необходимыми нам функциональными возможностями, но не под вывеской «модуль wiki». Мы можем использовать любые инструменты, предоставляемые Drupal, а не только те, что носят очевидные названия. Ядро системы Drupal уже содержит множество встроенных возможностей «wiki», поэтому нам достаточно добавить лишь несколько дополнительных штрихов, чтобы привнести в наш сайт то, что люди ожидают получить от wiki.

Простота форматирования текста

Одна из наиболее известных особенностей wiki-сайтов заключается в простоте форматирования, без использования разметки HTML. Форматирование в мире wiki обычно выполняется с помощью «разметки wiki», которая позволяет форматировать несложный текст подобно HTML, но намного проще. Одним из примеров популярных инструментов преобразования текста в разметку HTML является модуль Markdown (за дополнительной информацией обращайтесь по адресу *http://daringfireball.net/projects/markdown/*), позволяющий превращать простой текст в эквивалент на языке разметки HTML.

Для иллюстрации ниже приводится пример предложения, написанного на языке разметки HTML и на языке разметки Markdown:

I am a sentence with **bold** text and a link to another page.

Оба следующих примера воспроизводят фрагмент текста жирным шрифтом и гиперссылку с текстом «link to another page», ссылающуюся на страницу «page2»:

HTML

```
I am a sentence with <strong>bold</strong> text and a
<a href="/page2">link to another page</a>.
```

Markdown

```
I am a sentence with **bold** text and a [link to another page](/page2).
```

Как видно из этих двух примеров, упрощенный синтаксис совсем не страшен для тех, кто прежде никогда не пользовался языком HTML. Еще одно преимущество упрощенной разметки состоит в том, что, когда потребуется прочитать исходный текст страницы, синтаксис ее будет ближе к обычному тексту и потому будет проще восприниматься.

Возможность использования определенной разметки достигается за счет использования дополнительных модулей, расширяющих встроенную систему фильтров, о которой рассказывалось в главе 2. Существует достаточно широкое разнообразие вариантов разметки, поэтому ваш выбор в значительной степени может зависеть от личных предпочтений и познаний. На нашем сайте мы будем использовать модуль фильтра Markdown, но существует еще целое множество других модулей, включая BBcode (*http://drupal.org/project/bbcode*), Textile (*http://drupal.org/project/textile*) и многие другие. Полный перечень фильтров можно найти по адресу *http://drupal.org/project/Modules/category/63*.

Простота создания ссылок

Система фильтров может также использоваться для автоматического создания ссылок, то есть от нас не требуется знать и вводить адрес URL страницы назначения.

Фильтр Freelinking позволяет легко создавать ссылки на другие страницы сайта, даже если они еще не были созданы. Если ссылка указывает на несуществующую страницу, щелчок на ней будет перенаправлять вас на страницу с формой создания страницы, в которой поле заголовка уже будет заполнено, что существенно упрощает добавление новых страниц и устраняет необходимость щелкать по пунктам различных меню.

Отслеживание изменений

Нам также необходима возможность отслеживать все изменения в содержимом, поэтому мы добавим ее в наш список необходимых модулей. Drupal имеет встроенную систему редакций, которая выполняет всю основную работу, связанную с отслеживанием изменений в тексте и их

отменой. Для расширения этой системы и обеспечения возможности получения срезов между изменениями мы будем использовать модуль Diff.

Удобочитаемые адреса URL

Встроенный модуль Path позволяет получить удобочитаемые адреса URL, но он требует, чтобы мы вручную вводили и изменяли эти адреса. Задача автоматизации этого процесса решена в модуле Pathauto, который позволяет определять стандартные шаблоны слов для автоматической замены всех адресов URL при создании и изменении содержимого. Модуль Pathauto можно настроить так, чтобы он автоматически изменял пути к записям в блогах, такие как *http://www.example.com/2009/04/23/this-is-my-blog-entry-title*, и все пути к страницам с профилями пользователей, такие как *http://www.example.com/members/admin*.

Список изменений

Наше последнее требование, наличие списка изменений для всего сайта, легко может быть удовлетворено с помощью модуля Views, который обсуждался в главе 3. Мы создадим настраиваемый список материалов, подвергшихся изменению, который позволит членам ассоциации легко обнаруживать то, над чем работали другие.

Практика: первые шаги

В этом разделе рассматриваются основы, знание которых потребуется в остальных практических разделах главы.

Создание типа содержимого Wiki

Первое, что необходимо сделать, это убедиться в наличии необходимых типов содержимого. Студенческим органам самоуправления требуются страницы, доступные для изменения только модераторам, а также основное содержимое сайта, доступное для редактирования всем членам самоуправления. Чтобы различать эти два случая и упростить управление доступом, нам потребуются два типа содержимого: Page (Страница) и Wiki page (Страница wiki).

Как уже говорилось в главе 2, в Drupal имеется два предопределенных типа содержимого, Page и Story, из чего следует, что тип Page у нас уже имеется. Нам не потребуется тип Story, поэтому, вместо того чтобы создавать новый тип для wiki, мы можем приспособить тип Story под наши нужды.

Точно так же можно было бы удалить тип Story и создать новый тип Wiki page (Страница wiki). И даже оставить тип Story, чтобы использовать его для других нужд.

Перейдите на страницу Administer (Администрирование)→Content management (Содержание)→Content types (Типы материалов) (*admin/content/types*), и вы увидите существующие типы Page (Страница) и Story (Статья). Щелкните на ссылке edit (изменить) для типа Story и внесите изменения в соответствии с табл. 5.1, как показано на рис. 5.2. Щелкните на кнопке Save content type (Сохранить тип содержимого), чтобы создать новый тип содержимого.

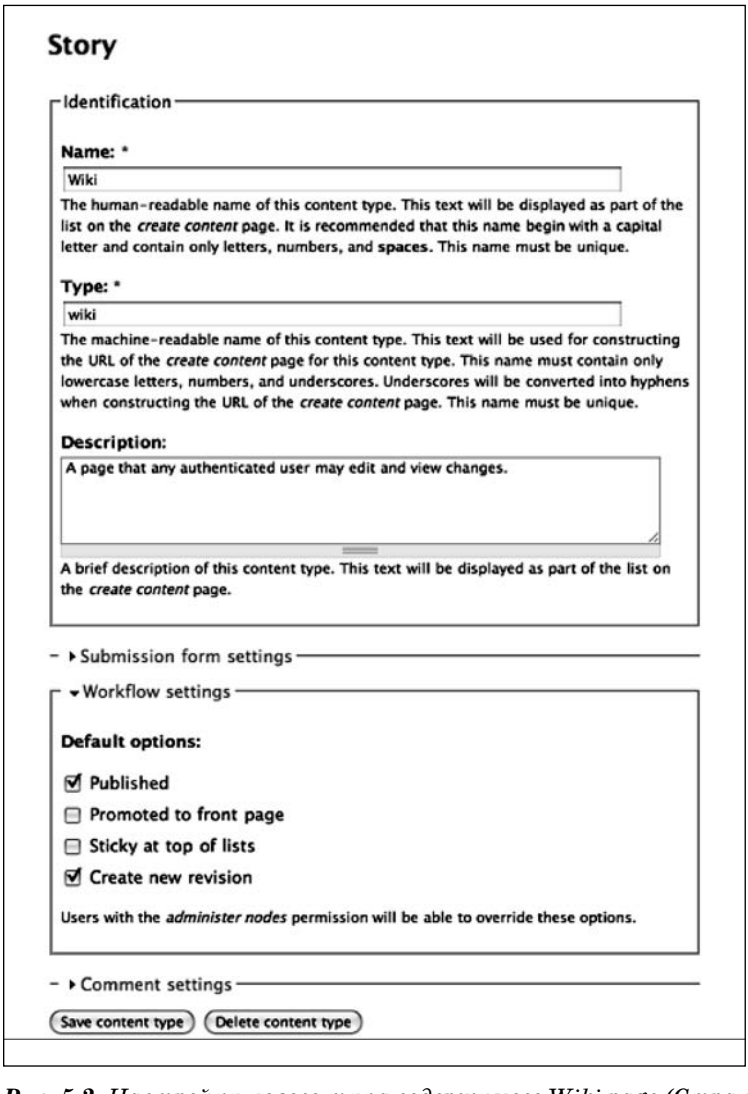


Рис. 5.2. *Настройки нового типа содержимого Wiki page (Страница wiki)*

Таблица 5.1. Настройки типа содержимого Wiki page (Страница wiki)

Параметр настройки	Значение
Name (Имя)	Wiki page (Страница wiki)
Type (Тип)	wiki
Description (Описание)	A page that any authenticated user may edit and view changes (Страница, изменять и просматривать изменения в которой может любой зарегистрированный пользователь)
Workflow settings (Процесс): Default options (Параметры по умолчанию)	Снять флажок Promoted to front page (Помещать на главную) Отметить флажок Create new revision (Создавать новую редакцию)

Удаление информации об авторе при отображении

Так как все содержимое сайта будет создаваться коллективно, нам не требуется отображать информацию «о пользователе, создавшем материал» (так как неизбежно любой материал не будет принадлежать одному автору). Чтобы удалить эту информацию из типа содержимого Wiki page (Страница wiki), необходимо ненадолго зайти в раздел настройки темы оформления:

1. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Themes (Темы оформления) (*admin/build/themes*).
2. Затем щелкните на вкладке Configure (Настроить) (*admin/build/themes/settings*).
3. Снимите флажок Wiki page (Страница wiki) в разделе Display post information on (Отображать информацию о сообщении).
4. Щелкните на кнопке Save configuration (Сохранить конфигурацию), чтобы сохранить изменения.

Настройка прав доступа

Нам необходимо, чтобы любой, кто обладает учетной записью, имел возможность создавать и редактировать содержимое wiki, но только отдельные должностные лица могли создавать или изменять обычные страницы на сайте.

Перейдите на страницу Administer (Администрирование)→User management (Управление пользователями)→Permissions (Разрешения) (*admin/user/permissions*) и выполните настройки в соответствии с табл. 5.2. Об-

ратите внимание, что мы не устанавливаем разрешение «edit own X content» («изменять собственное содержимое X»), потому что нам требуется, чтобы наши пользователи могли изменять *любое* содержимое, а не только свое собственное. По завершении щелкните на кнопке Save permissions (Сохранить права доступа).

Таблица 5.2. Разрешения на доступ к содержимому

Разрешение: модуль node	**anonymous user (анонимный пользователь)**	**authenticated user (зарегистрированный пользователь)**	**editor (редактор)**	**site administrator (администратор сайта)**
create page content (создавать материал типа page)			отмечен	отмечен
create wiki content (создавать материал типа wiki)		отмечен		
edit any page content (изменять любой материал типа page)			отмечен	отмечен
edit any wiki content (изменять любой материал типа wiki)		отмечен		

Практика: формат ввода wiki

Как вы наверняка помните по главе 2, где были представлены форматы ввода и фильтры, перед отображением все содержимое передается через фильтры, включенные в указанный формат ввода. Пользователи вводят любой бред, какой только пожелают, а затем Drupal фильтрует и изменяет этот текст при выводе.

Одно из требований к нашему сайту заключается в обеспечении простого способа форматирования и добавления ссылок. Фильтры Markdown и Freelinking обеспечивают такую возможность. Для нашего сайта wiki мы создадим новый формат ввода, который будет использовать несколько базовых фильтров, а также ряд наших собственных. В разделах ниже описываются следующие шаги:

1. Настройка фильтров.
2. Создание формата ввода wiki.
3. Настройка разрешений для формата.
4. Добавление содержимого.

Настройка фильтров

1. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Modules (Модули) (*admin/build/modules*) и включите следующие модули:
   - Input filters (Фильтры ввода)
     – Markdown
   - Other (Другой)
     – Freelinking
2. Перейдите на страницу Administer (Администрирование)→Site configuration (Настройка сайта)→Freelinking (*admin/settings/freelinking*) и измените значение параметра Default for new content (По умолчанию для нового содержимого) на Wiki page (Страница wiki), чтобы ссылки на несуществующие материалы автоматически создавали страницы типа wiki. Эти настройки изображены на рис. 5.3. Мы делаем это потому, что не все пользователи имеют разрешение создавать новые материалы типа Page (Страница) (в противоположность материалам типа Wiki page (Страница wiki)). (Обратите внимание, что фильтр Markdown не требует каких-либо дополнительных настроек.)

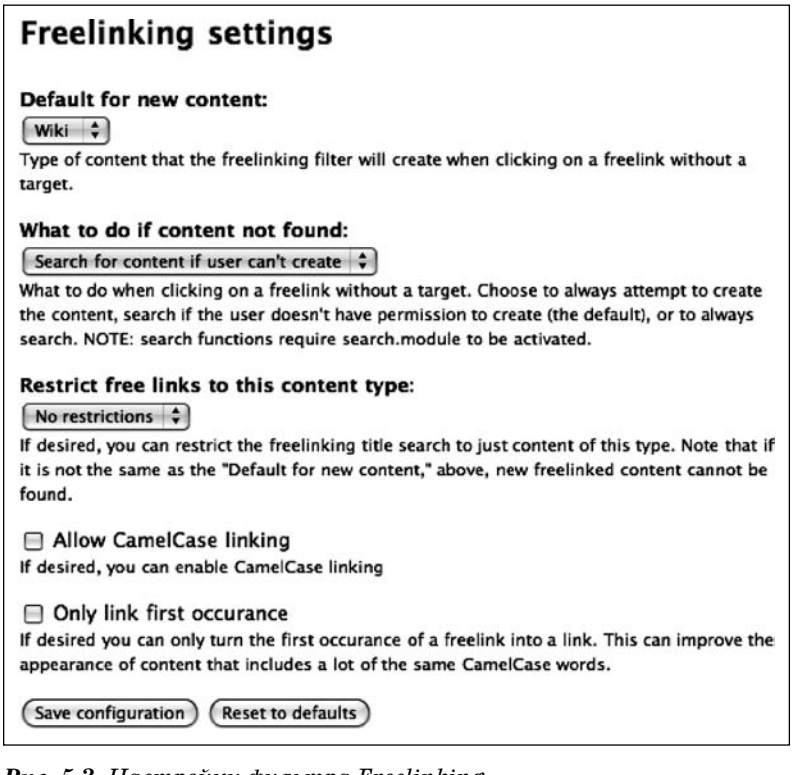


Рис. 5.3. *Настройки фильтра Freelinking*

3. Щелкните на кнопке Save configuration (Сохранить конфигурацию).
4. Перейдите на страницу Administer (Администрирование)→User management (Управление пользователями)→Permissions (Разрешения) (*admin/user/permissions*) и установите разрешения в соответствии с табл. 5.3. По завершении щелкните на кнопке Save permissions (Сохранить права доступа).

Таблица 5.3. Разрешения для модуля Freelinking

Разрешение: модуль freelinking	**anonymous user (анонимный пользователь)**	**authenticated user (зарегистрированный пользователь)**	**editor (редактор)**	**site administrator (администратор сайта)**
access freelinking list (доступ к списку freelinking)		отмечен		

Что такое ссылки в стиле CamelCase?

Для создания ссылок фильтр Freelinking по умолчанию использует обычную разметку, которая требует заключать ссылки в двойные квадратные скобки: [[имя ссылки]]. Более старый метод создания ссылок в разметке wiki заключается в том, чтобы просто записать ссылку в стиле CamelCase, где все слова объединяются вместе и первая буква каждого слова записывается в верхнем регистре (например, LinkName), без дополнительной разметки. Синтаксис Freelinking с квадратными скобками был введен для устранения некоторых проблем, связанных со стилем CamelCase, наиболее заметной из которых является невозможность использования названий ссылок, состоящих из единственного слова, без вставки заглавных букв в середине слова (WiKi), что делает такие ссылки более тяжелыми для восприятия. При использовании модуля Freelinking вы можете применять оба способа. Если у вас имеется желание разрешить создание ссылок в стиле CamelCase, отметьте соответствующий флажок на странице с настройками модуля Freelinking.

Вы можете заметить, что модуль Freelinking добавляет новый элемент Freelinks в навигационное меню. Сейчас на странице, к которой ведет эта ссылка, ничего нет, но по мере добавления ссылок в содержимое, использующее этот фильтр, на этой странице будут перечислены все новые ссылки. Так как у нас имеется возможность создавать ссылки на несуществующие страницы, этот список позволит нам быстро получить перечень страниц, которые требуется создать. Эта особенность будет полезна членам SGA, участвующим в наполнении сайта.

Создание формата ввода wiki

Два модуля, включенных нами, добавляют несколько новых фильтров, которые мы можем использовать. Мы можем добавлять их в существующие форматы ввода (такие как Filtered HTML) или на их основе создавать совершенно новые форматы ввода. Мы создадим новый формат, чтобы в случае необходимости у нас оставалась возможность вернуться к оригинальному формату Filtered HTML. Формат ввода – это просто упорядоченная коллекция фильтров (фильтры получают текст и преобразуют его, прежде чем он будет отображен).

В первую очередь мы должны выбрать, какие фильтры потребуются, а затем определить порядок их следования. Содержимое будет передаваться каждому фильтру в том порядке, в каком они перечислены. Если невнимательно отнестись к порядку следования фильтров, может получиться так, что один фильтр будет затирать результаты работы другого фильтра, и мы не получим результат, который ожидали. Ниже описаны шаги по созданию формата ввода:

1. Перейдите на страницу Administer (Администрирование)→Site configuration (Настройка сайта)→Input formats (Форматы ввода) (*admin/settings/filters*). Щелкните на вкладке Add input format (Добавить формат ввода) и выполните следующие настройки:
   - В параметре Name (Имя) установите значение Wiki.
   - Отметьте: HTML corrector (HTML-корректор), HTML filter (Фильтр HTML), Line break converter (Преобразователь переводов строки), Markdown, URL filter (Фильтр URL) и freelinking filter и сохраните настройки.
2. Вернувшись на страницу Input formats (Форматы ввода), щелкните на ссылке configure (настроить) для нового формата Wiki.
3. Щелкните на вкладке Rearrange (Перестроить) и перетащите мышью фильтры freelinking filter и Markdown в начало списка, как показано на рис. 5.4.
4. По окончании щелкните на кнопке Save configuration (Сохранить конфигурацию).

Порядок следования фильтров очень важен. Фильтры Markdown и Freelinking должны идти первыми, потому что они оба преобразуют обычный текст в разметку HTML. Для обеспечения безопасности важно, чтобы следующим в списке был фильтр HTML, потому что он ограничивает перечень тегов HTML, которые может вставлять пользователь. Поместив наши новые фильтры первыми, мы позволяем фильтру HTML проверить результаты их работы и устранить все нежелательное, прежде чем содержимое будет отображено.

Если вам придется столкнуться с невероятной ситуацией, когда ожидаемое преобразование текста не выполняется, попробуйте переупорядочить фильтры в данном формате ввода, так как действие одних фильтров может отменять действие других.

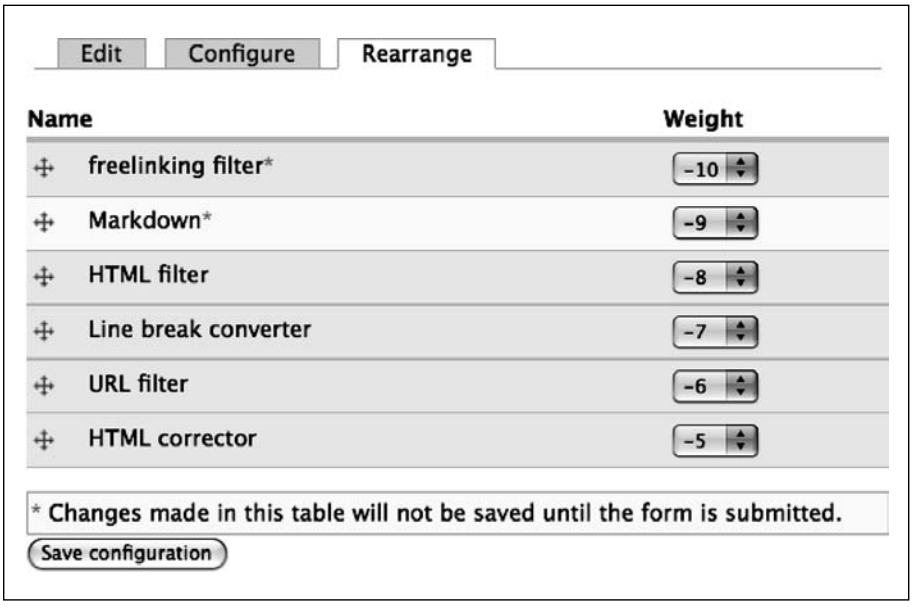


Рис. 5.4. *Порядок следования фильтров для формата Wiki*

Настройка разрешений формата

Так как нам необходимо, чтобы любой пользователь мог использовать этот новый формат, мы назначим его форматом по умолчанию для всего сайта.

Перейдите на страницу Administer (Администрирование)→Site configuration (Настройка сайта)→Input formats (Форматы ввода) (*admin/settings/filters*), отметьте радиокнопку рядом с названием формата Wiki и щелкните на кнопке Set default format (Установить формат по умолчанию).

Теперь формат Filtered HTML (Фильтрованный HTML) больше не является форматом по умолчанию, а все, что раньше делал этот формат, теперь будет делать новый формат Wiki. Одновременное присутствие этих форматов не только не нужно, но и будет сбивать с толку, поэтому давайте изменим разрешения так, чтобы удалить формат Filtered HTML (Фильтрованный HTML) из доступных вариантов выбора.

На Input formats (Форматы ввода) щелкните на ссылке configure (настроить) для формата Filtered HTML (Фильтрованный HTML) и снимите флажки anonymous user (анонимный пользователь) и authenticated user (зарегистрированный пользователь) в разделе Roles (Роли). Затем щелкните на кнопке Save configuration (Сохранить конфигурацию).

По окончании страница с форматами ввода должна выглядеть, как изображено на рис. 5.5.

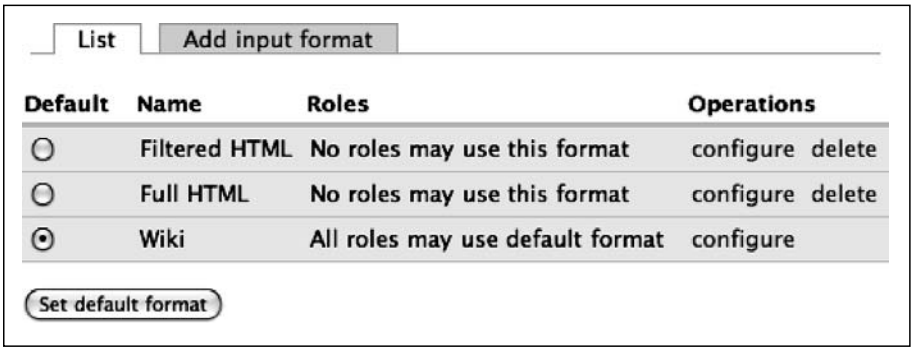


Рис. 5.5. *Список форматов ввода с форматом Wiki, выбранным как формат по умолчанию*

Добавление содержимого

Теперь добавим немного содержимого, для вывода которого используется новый формат ввода, и проверим его в действии. У нас уже имеется простая главная страница с текстом приветствия. Первое, что наверняка захотят увидеть новые пользователи, – это некоторые пояснения. Давайте добавим немного форматирования и создадим ссылку на новую страницу Help (Справка), которой еще не существует. Следует понимать, что главная страница представляет собой содержимое типа Page (Страница), а не Wiki page (Страница wiki). То есть редактировать эту страницу могут только пользователь *admin* (пользователь №1) и пользователи с ролью *editor* или *administrator*.

1. Перейдите на главную страницу и щелкните на вкладке Edit (Изменить).
2. Измените текст, чтобы он выглядел, как показано ниже:

```
Welcome to the **SGA wiki site**. This wiki is a central place for all
members to work together on internal documentation and planning.
There is a [[Help]] page that explains how to create a new page and add
formatting and links.
```

3. Так как формат ввода Wiki был создан и назначен форматом по умолчанию уже после создания этой страницы, нам также необходимо установить для этой страницы формат Wiki. Раскройте группу параметров Input format (Формат ввода) и отметьте радиокнопку Wiki, как показано на рис. 5.6.

   Сохранив страницу, мы должны увидеть новое содержимое, которое должно выглядеть, как изображено на рис. 5.7. Можно заметить, что теперь название сайта выводится жирным шрифтом (благодаря действию фильтра Markdown) и на странице присутствует ссылка Help (Справка) (благодаря действию фильтра Freelinking).

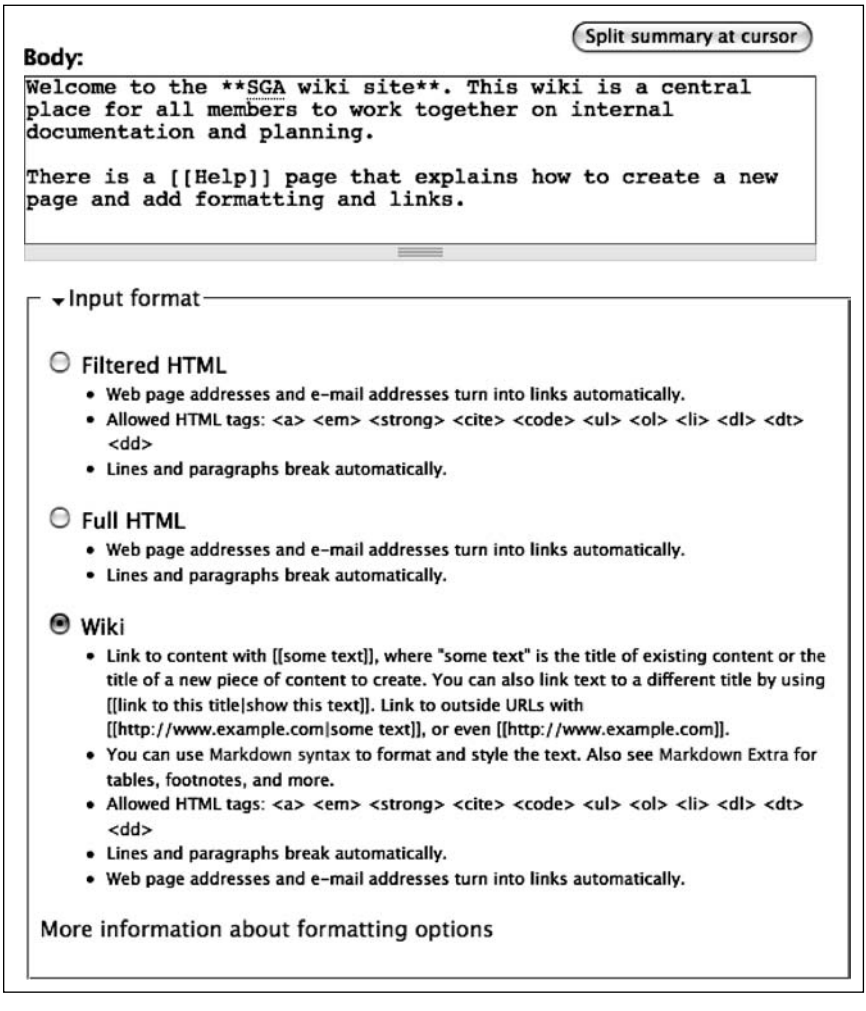


Рис. 5.6. *Настройка формата Wiki на главной странице*

4. Щелкните на ссылке Help (Справка) в тексте, и перед вами откроется новая страница с формой создания содержимого типа Wiki page (Страница wiki), с названием «Help» в поле Title (Заголовок), как показано на рис. 5.8.

   Вспомните, что при настройке модуля Freelinking мы решили, что любое содержимое, созданное в результате щелчка на ссылке, сгенерированной фильтром Freelinking, должно иметь тип Wiki page (Страница wiki). Это обеспечивает немедленную доступность всех страниц для редактирования всем пользователям, имеющим учетную запись на сайте.

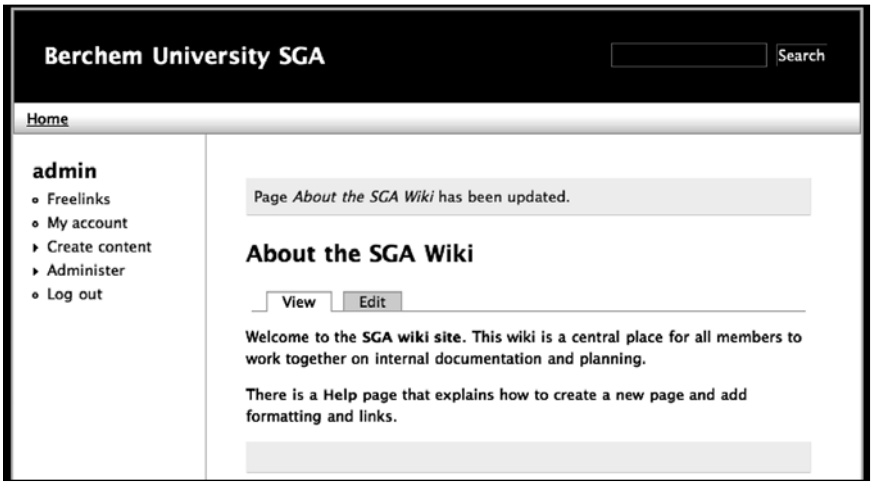


Рис. 5.7. *Новая страница About (О нас), для вывода которой используется формат ввода Wiki*

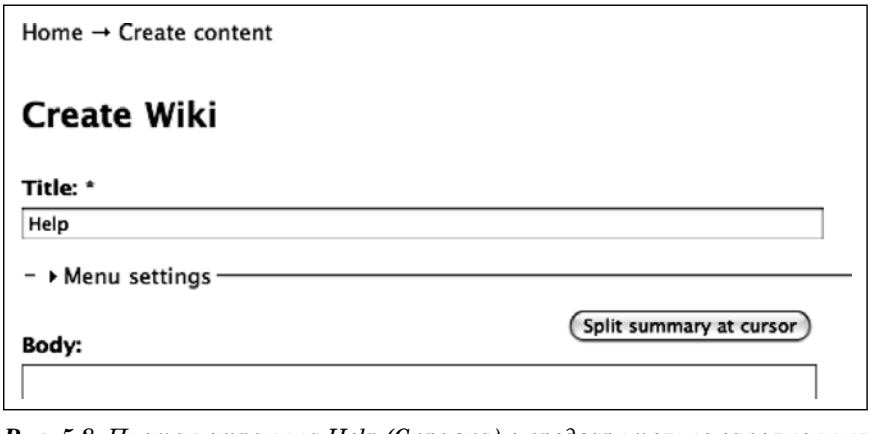


Рис. 5.8. *Пустая страница Help (Справка) с предварительно заполненным полем Title (Заголовок)*

5. Добавьте для наших пользователей некоторую справочную информацию со следующим текстом в поле Body (Содержание) и щелкните на кнопке Save (Сохранить), чтобы сохранить ее:

```
You can quickly create a new Wiki page by clicking Create content >
Wiki page from the navigation menu.

You can read more about this site's purpose on the [[About the SGA Wiki]] page.
```

   Вы увидите, что появилась новая страница Help (Справка) и что текст, который мы поместили в двойные квадратные скобки, образует ссылку на страницу About (О нас).

В центре внимания: модуль Pathauto

В главе 2 вы познакомились с понятием пути в Drupal и как использовать «чистые» адреса URL. Одна из причин использования таких адресов URL заключается в том, что они выглядят более красиво. (Для справки, «очистка» адреса URL состоит из удаления из URL части «?q=».) Это несколько улучшает ситуацию, но ненамного. Адрес URL, в котором указан путь «node/123», на практике почти ничего не говорит человеку или поисковой системе о самой странице. Разве не было бы лучше, если бы адрес URL содержал, например, такой путь, как «about-us»? Такой адрес проще запоминается, и добавление осмысленных ключевых слов в URL положительно сказывается на поисковой оптимизации. Таким образом, есть выигрыш от использования осмысленных путей даже без использования чистых URL.

Мы уже рассматривали базовый модуль Path в главе 2. При включении этот модуль добавляет в форму создания/изменения узла новую группу параметров настройки, которая называется URL path settings (Настройки адресов), как показано на рис. 5.9. Развернув эту группу, вы увидите поле для ввода альтернативного имени этого узла в пути (*синоним*). Имя, введенное в это поле, будет использоваться взамен пути, генерируемого системой Drupal, то есть вместо части адреса URL, следующей за *http://example.com/* (или *http://example.com/?q=*, если вы не используете чистые адреса URL).

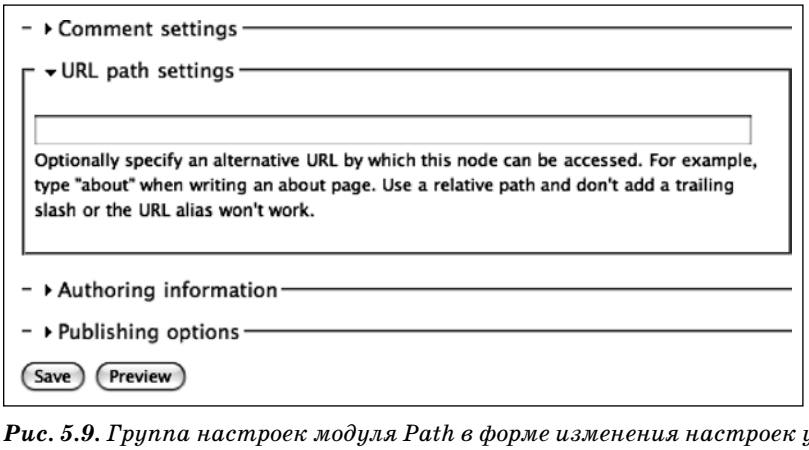


Рис. 5.9. *Группа настроек модуля Path в форме изменения настроек узла*

Это поле оказывает существенную помощь, но при создании большого количества страниц может быть достаточно утомительно вводить все эти имена вручную. Кроме того, при наличии большого числа пользователей, наделенных правом создавать содержимое, вам придется удостовериться, что они все правильно понимают и используют непротиворечивые имена, а это может стать настоящей головной болью для администратора. Еще один недостаток модуля Path заключается в том, что он

позволяет указывать синонимы только для узлов. А как быть с профилями пользователей или с путями словарей?

Как это нередко бывает, сторонние проекты предоставляют модули, позволяющие решить эту задачу. Представляем вам модуль Pathauto. Как позволяет предположить его название, он автоматически создает синонимы путей для узлов, таксономии и профилей пользователей. Модуль Pathauto опирается на базовый модуль Path и еще на один модуль, с именем Token, описание которого приводится во врезке.

Что такое лексема (token)?

Лексемы – это заполнители. Например, лексема [yyyy] представляет номер года, состоящий из четырех цифр, а лексема [authorname] представляет имя пользователя автора узла. Возможность использования лексем в Drupal обеспечивается модулем Token. Модуль Token отличается от большинства других модулей, используемых на сайте, тем, что его функциональность не заметна явно для администраторов или пользователей сайта. Она необходима для работы других модулей, таких как Pathauto, реализующих интерфейс. Модуль Token создает централизованный репозиторий заполнителей. Модули, обладающие данными (такие как базовый модуль Node, который знает дату создания узла), могут с помощью модуля Token обеспечить доступность своей информации. Такие модули являются своего рода «поставщиками». Модули, которые нуждаются в этой информации, такие как Pathauto, могут обращаться к модулю Token, чтобы получить список того, что доступно. Эти модули являются своего рода «потребителями». Модуль Token действует, как своеобразный магазин, играя роль посредника между поставщиками и потребителями.

Сам модуль Token также предоставляет несколько простых заполнителей, доступных по умолчанию, данные для которых поставляются встроенными модулями Drupal. Наиболее заметными из них являются типы содержимого (модуль Node), словари (модуль Taxonomy) и пользователи (модуль User). Сторонние модули могут выступать не только в роли «потребителей», но и создавать новые заполнители, представляющие данные из этих модулей. Им нужно просто на языке модуля Token сказать: «Эй, у меня есть элемент данных xyz, и вы можете сообщить об этом другим, предложив им использовать заполнитель [xyz]».

Для настройки шаблонов адресов URL модуль Pathauto использует комбинацию простого текста и «лексем». Например, можно создать такой шаблон пути ко всем новым страницам в блоге: «blog/[yyyy]/[month]/[date]/[title-raw]», который автоматически будет превращаться в пути,

такие как «blog/2007/june/08/my-first-blog-post». На рис. 5.10 показан параметр настройки Automatic alias (Автоматическая подстановка синонима) модуля Pathauto для страницы wiki, озаглавленной, как «Help» (Справка), где используется шаблон «wiki/[title-raw]».

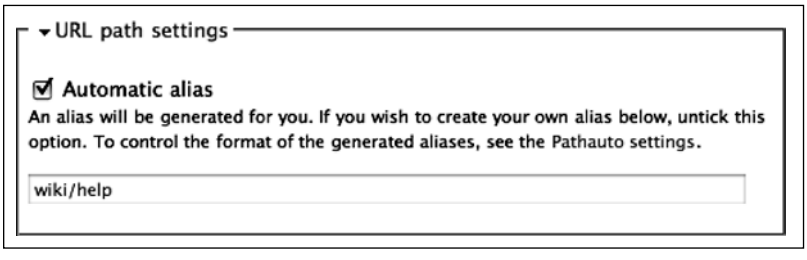


Рис. 5.10. *Версия настройки пути модуля Pathauto в форме изменения узла, с автоматическим заполнением пути*

Элементы шаблона, заключенные в квадратные скобки, – это заполнители, предназначенные для модуля Token, упомянутого выше. Существует возможность создавать подобные шаблоны для любого типа содержимого или словаря. Кроме того, в модуле Pathauto имеются настраиваемые шаблоны, которые будут применяться по умолчанию, если вы не определите свои собственные. Модуль Pathauto позволяет настраивать такие параметры, как предельная длина псевдонима, какие разделители следует использовать вместо пробелов и знаков пунктуации и какие стандартные сокращения требуется удалять (напр., и, в, и пр., и т. д.). Помимо автоматического образования синонимов при создании нового содержимого модуль Pathauto может также обновлять существующее содержимое, чтобы один и тот же шаблон использовался по всему сайту, даже если это содержимое было создано до подключения модуля Pathauto.

При использовании модуля Pathauto очень важно определиться, как вы собираетесь управлять изменением синонимов. Так как синоним создается на основе информации, которую модуль Pathauto получает из содержимого, то при изменении содержимого могут изменяться и синонимы. При изменении содержимого вы можете указать, как эти изменения должны отражаться на синонимах:

- Do nothing. Leave the old alias intact. (Не изменять. Оставить прежний синоним в неприкосновенности.)
- Create a new alias. Leave the existing alias functioning. (Создать новый синоним и оставить прежний функционирующим.)
- Create a new alias. Delete the old alias. (Создать новый синоним и удалить прежний. Используется по умолчанию.)

Для различных сайтов могут быть свои причины выбора того или иного способа управления синонимами. По умолчанию будет создаваться новый синоним и удаляться старый, чтобы синонимы всегда соответство-

вали содержимому. Этот вариант может приводить к появлению «битых» ссылок. Битая ссылка может получиться, например, в такой ситуации: на вашем сайте имеется страница с определенным адресом URL, таким как *http://example.com/about*, а на другом сайте создается ссылка на этот адрес. Если вы замените этот адрес URL на *http://example.com/about-us* и удалите прежний, все внешние ссылки на эту страницу перестанут функционировать. Это и есть «битые» ссылки.

Второй вариант, когда создается новый синоним и сохраняется прежний, может показаться идеальным, потому что вы сможете получать доступ к содержимому по любому пути и проблемы битых ссылок просто не возникает. Но, ликвидируя проблему битых ссылок, этот вариант порождает другую проблему – поисковые системы будут «штрафовать» вас за наличие нескольких путей, ведущих на одну и ту же страницу, потому что они полагают, что вы можете пытаться подтасовать результаты поиска. Один из способов обойти эту проблему состоит в том, чтобы в паре с модулем Pathauto задействовать модуль Path Redirect (*http://drupal.org/project/path_redirect*). При использовании модуля Path Redirect к первым трем вариантам настройки синонимов добавляется четвертый: Create a new alias. Redirect from old alias (Создать новый синоним. Перенаправлять со старого синонима). При использовании этого варианта все обращения по старому адресу URL автоматически будут перенаправляться по новому адресу URL.

Просмотрите настройки модуля Pathauto и поиграйте с ними, пока сайт находится в разработке, чтобы понять, какой из вариантов лучше подходит для вас. После того как сайт будет запущен и люди начнут пользоваться им, следует избегать вносить серьезные изменения в настройки модуля Pathauto, так как пользователи могут полагаться на определенное поведение адресов URL.

Практика: модуль Pathauto

Чтобы получить эти удобочитаемые адреса URL на своем сайте, мы будем использовать модуль Pathauto, который, как было показано в предыдущем разделе, опирается на модуль Token, а также на встроенный модуль Path.

Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Modules (Модули) (*admin/build/modules*) и включите следующие модули:

- Core – optional (Ядро – необязательные)
  - Path
- Other (Другой)
  - Pathauto
  - Token

Настройка параметров

Страница настройки модуля Pathauto содержит достаточно большое число параметров и может выглядеть устрашающе. К счастью, большая часть параметров имеет значения по умолчанию, пригодные для использования на большинстве сайтов, что делает работу по настройке намного проще, чем может показаться на первый взгляд.

Лексемы модуля Pathauto, оканчивающиеся на «–raw»

Если просмотреть лексемы, принадлежащие модулю Pathauto, можно заметить, что существуют лексемы с похожими именами – одна из которых является простым именем, а другая с тем же именем, к которому добавлен суффикс «-raw», например, [title] и [title-raw]. Обратите внимание на предупреждение в справочном тексте для лексем с суффиксом «-raw». Лексемы без этого суффикса фильтруются с целью устранения вредоносного содержимого. Лексемы с суффиксом «-raw» фильтрации не подвергаются, поэтому, казалось бы, вам следует избегать их использования, однако для них модуль Path, встроенный в Drupal, использует собственные фильтры, обеспечивая безопасность этих лексем.

Поэтому при использовании модуля Pathauto не только безопасно применять лексемы «-raw», но и рекомендуется. Модулю Pathauto необходима полная, неотфильтрованная информация, чтобы обеспечить корректную замену знаков пунктуации при создании нового синонима.

1. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→URL aliases (Синонимы) (*admin/build/path*) и щелкните на вкладке Automated alias settings (Настройки автоматических синонимов) (*admin/build/path/pathauto*). Добавьте следующие настройки:
   - Разверните группу настроек Node path settings (Настройки адресов узлов) и в поле Pattern for all Wiki page paths (Шаблон путей к страницам типа Wiki) введите «wiki/[title-raw]», как показано на рис. 5.11.
   - Разверните группу настроек User path settings (Настройки адресов пользователей) и измените значение «users/[user-raw]» на «members/[user-raw]».

Эти настройки будут применяться к вновь создаваемому содержимому, но нам необходимо принять во внимание содержимое, которое уже существует. Сейчас наша недавно созданная страница Help (Справка) имеет адрес URL *http://example.com/node/2*. Нам необходимо применить к этой странице тот же самый шаблон, который будет применяться ко

всем новым страницам. Для этого надо предписать модулю Pathauto выполнить массовое обновление существующих путей:

2. Снова разверните группу настроек Node path settings (Настройки адресов узлов) и в самом низу отметьте флажок Bulk generate aliases for nodes that are not aliased (Выполнить массовое создание синонимов для узлов, не имеющих синонимов).

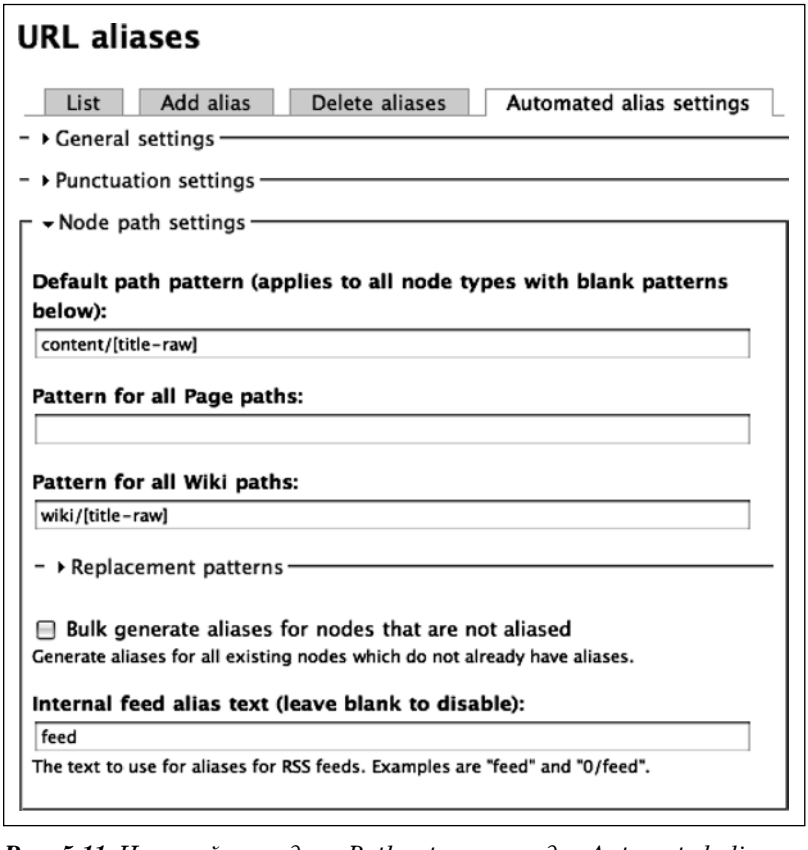


Рис. 5.11. *Настройки модуля Pathauto на вкладке Automated alias (Настройки автоматических синонимов) на странице настройки адресов в административном разделе*

3. Точно так же распахните группу настроек User path settings (Настройки адресов пользователей) и отметьте тот же самый флажок, как изображено на рис. 5.12.

Сохранив настройки щелчком на кнопке Save configuration (Сохранить конфигурацию), убедитесь, что теперь ранее созданная страница Help (Справка) имеет адрес URL *http://example.com/wiki/help*, а при переходе по ссылке My Account (Мой профиль) можно заметить, что теперь адрес URL имеет вид *http://example.com/members/admin*.

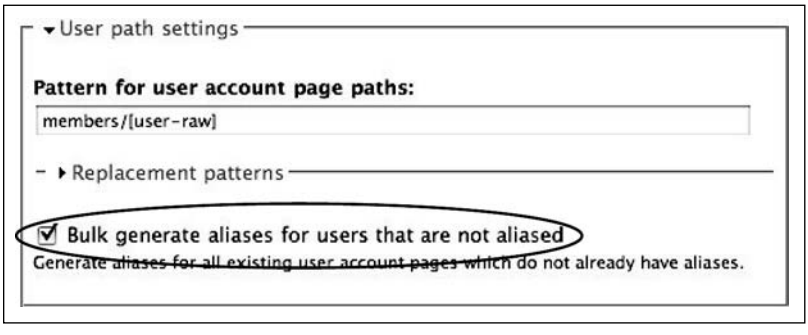


Рис. 5.12. *Выполнить массовое создание синонимов для пользователей, не имеющих синонимов*

В центре внимания: отслеживание редакций в Drupal

У любого, кому приходилось создавать тексты в сети или в локальном текстовом процессоре, бывали моменты, когда сразу после щелчка на кнопке Submit (Отправить) или Save… (Сохранить) появлялось желание отменить последнее действие. Или, например, вы что-то старательно создавали, затем передали другим для доработки, а они все испортили. В этот момент вам наверняка хотелось бы иметь предыдущую версию вашей работы. К счастью, в системе Drupal такая возможность включена прямо в ядро. По мере внесения изменений в содержимое в течение длительного времени Drupal сохраняет все версии содержимого, поэтому вы можете вернуться обратно к той версии, которая вас больше всего устраивает. Каждое сохраненное изменение в Drupal называется *редакцией*.

Каждый раз, когда выполняется сохранение узла, вместо того чтобы перезаписывать существующее содержимое, Drupal сохраняет новую копию узла. Однако такой порядок не является действующим по умолчанию – вам нужно сообщить системе Drupal, что вы желаете сохранить копию. Пользователь, обладающий разрешением administer nodes (управлять узлами), может активировать механизм сохранения редакций для каждого отдельного случая, включив флажок Create new revision (Создавать новую редакцию) в группе параметров Publishing options (Настройки публикации). Такой вариант отлично подходит для случаев, когда нежелательно сохранять в своей базе данных каждую копию содержимого и когда надлежащее разрешение выдано всем, кому оно может быть нужно. Имеется также возможность задействовать механизм редакций для отдельных типов содержимого. При таком подходе вам не потребуется помнить о необходимости устанавливать флажок при редактировании содержимого и не потребуется давать пользователям больше прав, чем могло бы потребоваться.

В дополнение к возможности сохранять различные версии текста вы также можете к каждой редакции добавлять примечание с помощью поля ввода Log message (Сообщение в системный журнал). Это очень удобно для описания изменений и причин, их вызвавших. Причины, вызвавшие необходимость изменений, могут казаться очевидными в момент их внесения, но это не означает, что они будут очевидны для других, работающих вместе с вами над этим же документом, и даже для вас эти причины могут не выглядеть такими очевидными спустя полгода. Никогда не лишне оставить примечание, касающееся изменений, – чтобы вам и другим было легко работать в дальнейшем.

После создания новой редакции узла рядом со стандартными вкладками View (Просмотреть) и Edit (Изменить) появится вкладка Revisions (Редакции). На вкладке Revisions (Редакции) можно просмотреть предыдущие версии, для каждой из которых имеется ссылка, содержащая информацию о дате и об авторе изменения, а также примечание, если оно имеется, которое было введено при сохранении. Здесь же можно выполнить возврат к определенной редакции, которая заменит содержимое на вкладке View (Просмотреть). При этом содержимое не будет удалено, а просто будет создана еще одна редакция, к которой можно будет вернуться позднее, если вы передумаете. Если вы действительно захотите удалить редакцию из базы данных, для этого здесь же имеется удобная ссылка.

Практика: редакции и модуль Diff

Давайте теперь посмотрим, как работает система редакций. Для этого нам потребуется отметить несколько флажков и затем внести какие-либо изменения в содержимое. Как только вы узнаете, как действует базовый механизм редакций в ядре Drupal, мы дополним его модулем Diff, сделав этот механизм еще более информативным.

Включение механизма редакций по умолчанию

Нам требуется сохранять все изменения в любом содержимом, будь то содержимое типа Wiki page или Page. Как уже упоминалось выше, по умолчанию механизм создания редакций отключен, но мы можем изменить это положение. Настройки публикации могут определяться для каждого типа содержимого по отдельности. Мы уже изменяли эти настройки при создании типа содержимого Wiki page, но для типа Page механизм редакций пока еще отключен. Давайте включим его, чтобы мы могли сохранять каждую версию:

1. Перейдите на страницу Administer (Администрирование)→Content management (Содержание)→Content types (Типы материалов) (*admin/content/types*) и щелкните на ссылке edit (изменить) для типа Page (Страница).

2. Прокрутите страницу вниз. В разделе Workflow settings: Default options (Процесс: Настройки по умолчанию) отметьте флажок Create new revision (Создавать новую редакцию).
3. Щелкните на кнопке Save content type (Сохранить тип содержимого), чтобы сохранить форму.

Настройка разрешений

Для сайта SGA нам необходимо, чтобы все пользователи имели возможность отслеживать действия друг друга и просматривать редакции. Кроме того, все они с достаточной степенью доверия относятся друг к другу, поэтому могут позволить любому отменять изменения в случае необходимости. Чтобы обеспечить все это, быстро наведаемся в раздел управления пользователями:

1. Перейдите на страницу Administer (Администрирование)→User management (Управление пользователями)→Permissions (Разрешения) (*admin/user/permissions*) и выполните настройки в соответствии с табл. 5.4.

Таблица 5.4. Настройка разрешений на доступ к редакциям

Разрешение: модуль node	**anonymous user (анонимный пользователь)**	**authenticated user (зарегистрированный пользователь)**	**editor (редактор)**	**site administrator (администратор сайта)**
revert revisions (переключаться между редакциями)		отмечен		
view revisions (просматривать редакции)		отмечен		

2. Щелкните на кнопке Save permissions (Сохранить права доступа), чтобы сохранить их.

Просмотр редакций и переключение между ними

Давайте внесем какие-нибудь изменения в содержимое и посмотрим, как действует механизм редакций:

1. Перейдите на страницу Help (Справка), созданную выше, и щелкните на вкладке Edit (Изменить).
2. Измените текст в поле Body (Содержимое), добавив ссылку на форму создания новых страниц wiki и удалив слово «Wiki», чтобы теперь текст выглядел как:

```
You can quickly create a new page by clicking “Create content >
Wiki page” from the navigation menu.
```

3. В поле Log message (Сообщение в системный журнал) в группе параметров Revision information (Информация о редакции) введите текст «Edited page creation sentence for clarity» (Отредактировано предложение о создании страницы, чтобы сделать его более понятным). Можно видеть, что флажок Create new revision (Создавать новую редакцию) отмечен по умолчанию.
4. После сохранения изменений для вашей страницы рядом с вкладками View (Просмотреть) и Edit (Изменить) появится новая вкладка Revisions (Редакции). Щелкните на вкладке Revisions (Редакции).
5. Щелкните на ссылке с самой ранней датой в колонке Revision (Редакция), и вы увидите самую первую версию страницы. Изменившийся заголовок страницы однозначно свидетельствует, что вы просматриваете редакцию страницы, время создания которой указано в заголовке, как показано на рис. 5.13.

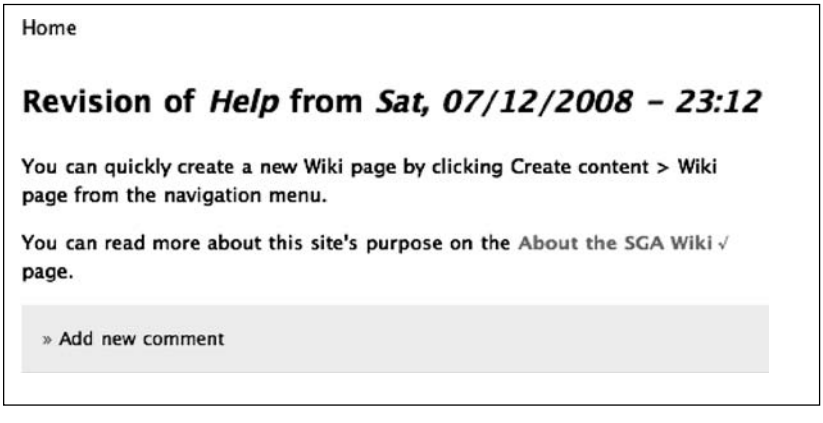


Рис. 5.13. *Просмотр редакции страницы Help (Справка)*

6. Щелкните на кнопке Back (Назад) броузера и вернитесь к списку редакций.
7. Щелкните на ссылке revert (вернуть) рядом с самой ранней записью и затем на кнопке Revert (Вернуть), чтобы подтвердить свое намерение.
8. Вы увидите, что в списке редакций появилась новая запись, которая показывает, какую редакцию мы скопировали. На рис. 5.14 изображен список редакций после изменений, сделанных нами. Если теперь щелкнуть на вкладке View (Просмотреть), вы увидите, что был возвращен оригинальный текст.

Использование модуля Diff

Мы уже можем просматривать список редакций нашего содержимого и переключаться между ревизиями, но членам SGA в действительно-

Home → Help

Wiki *Help* has been reverted back to the revision from *Sat, 07/12/2008 - 23:12.*

Revisions for *Help*

The revisions let you track differences between multiple versions of a post.

View | Edit | Revisions

Revision	Operations
07/12/2008 - 23:34 by admin Copy of the revision from *Sat, 07/12/2008 - 23:12.*	*current revision*
07/12/2008 - 23:26 by admin Edited page creation sentence for clarity	revert delete
07/12/2008 - 23:12 by admin	revert delete

Рис. 5.14. *Список всех редакций страницы Help (Справка)*

сти требуется иметь возможность просматривать различия между редакциями. Небольшие изменения на большой странице легко пропустить при попытке увидеть их, просматривая полный текст каждой редакции. Представляем вам модуль Diff. Этот модуль отлично интегрируется с существующей системой редакций; он позволяет нам получать представление, содержащее только те строки текста, которые изменились, и услужливо подсвечивает измененные слова. Любой, обладающий разрешением «view revisions» (просматривать редакции), сможет использовать модуль Diff.

Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Modules (Модули) (*admin/build/modules*) и включите модуль Diff в разделе Other (Другой).

Теперь вернитесь на страницу редакций, щелкнув на вкладке Revisions (Редакции) на странице Help (Справка), и вы увидите новую кнопку с надписью Show Diff (Показать изменения) и группу радиокнопок, по одной для каждой редакции. Чтобы увидеть разницу между двумя версиями, выберите радиокнопки для редакций, которые требуется сравнить, и щелкните на кнопке Show Diff (Показать изменения). На рис. 5.15 показано, как выглядит страница, отображающая изменения между редакциями.

Home

Revisions for *Help*

Sat, 07/12/2008 - 23:26 by admin	Sat, 07/12/2008 - 23:34 by admin
Edited page creation sentence for clarity	Copy of the revision from *Sat, 07/12/2008 - 23:12.*

< previous diff

Changes to *Body*

Line *1*	Line *1*
- You can quickly create a new page by clicking "Create content > Wiki page" from the navigation menu.	+ You can quickly create a new Wiki page by clicking Create content > Wiki page from the navigation menu.
You can read more about this site's purpose on the [[About the SGA Wiki]] page.	You can read more about this site's purpose on the [[About the SGA Wiki]] page.

Рис. 5.15. *Модуль Diff подсвечивает изменения между редакциями страницы Help (Справка)*

Практика: создание представлений New pages и Recent edits

В состав ядра Drupal входит модуль Tracker, который создает удобную страницу со списком последних обновленных материалов. Однако то, что предлагает эта страница, не совсем то, что требуется членам SGA. Модуль Tracker включает в список все новые материалы, включая изменения, а материалы, к которым были добавлены новые комментарии, считает «обновленными». Это означает, что значительная часть материалов, присутствующих в списке, в действительности не изменялась. Кроме того, в списке перемешаны вновь созданные и измененные материалы. Членам SGA хотелось бы иметь два списка: в одном отображались бы вновь созданные материалы, а во втором – только те материалы, которые были отредактированы.

Поскольку встроенный модуль Tracker не обеспечивает необходимой нам гибкости, для создания требуемых списков мы будем использовать модуль Views, представленный в главе 3. В состав модуля Views по умолчанию входит представление, которое дублирует список, создаваемый модулем Tracker. Мы будем использовать его как основу для двух списков, которые требуется создать: «Recent posts» (Последние материалы) и «Recent edits» (Последние изменения).

Список Recent posts

1. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Modules (Модули) (*admin/build/modules*) и включите следующие модули:
   - Views
     - Views
     - Views UI
2. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Views (*admin/build/views*) и щелкните на ссылке Enable (Включить) для представления Default Node view: tracker (Представление узла по умолчанию: tracker). Затем щелкните на ссылке Edit (Изменить) (*admin/build/views/edit/tracker*).
3. Выделите настройки отображения Defaults (По умолчанию).
4. Ограничим доступность этого представления только зарегистрированными пользователями, выполнив базовые настройки в соответствии с табл. 5.5 и щелкнув на кнопке Update (Обновить). На следующей странице с настройками выберите роль authenticated user (зарегистрированный пользователь) и щелкните на кнопке Update (Обновить) еще раз.

Таблица 5.5. Базовые настройки представления «Recent posts» (Последние материалы)

Defaults: Basic settings (По умолчанию: Базовые настройки)	**Значение**
Access (Доступ)	Type (Тип): By role (По роли)

5. Поскольку это представление должно отображать только новые материалы, в разделе Fields (Поля) щелкните на ссылке Node: Last comment time (Узел: Время последнего комментария) и щелкните на кнопке Remove (Удалить), чтобы удалить поле из представления.
6. Замените время последнего комментария временем создания материала, щелкнув на ярлыке с изображением знака плюс в группе Fields (Поля) и добавив поле Node: Post date (Узел: Дата). Запишите в настройку Label (Название) текст «Post date» (Дата создания).
7. Поскольку список последних материалов должен содержать все новые материалы, появившиеся на сайте, независимо от пользователя, в разделе Arguments (Аргументы) щелкните на аргументе User: Uid (Пользователь: Uid) и затем на кнопке Remove (Удалить).
8. Учитывая изменения в наборе полей, отсортируем список материалов по дате создания, а не по времени последнего комментария. В разделе Sort criteria (Критерий сортировки) щелкните на поле Node: Last comment time (Узел: Время последнего комментария) и затем на кнопке Remove (Удалить), чтобы удалить его из представления.

9. Щелкните на ярлыке с изображением знака плюс в разделе Sort criteria (Критерий сортировки) и отметьте флажок Node: Post date (Узел: Дата). Переключите порядок сортировки в значение Descending (По убыванию).
10. По выполнении этих шагов настройки представления должны выглядеть, как показано на рис. 5.16. Щелкните на кнопке Save (Сохранить).
11. Щелкните на ссылке View «Page» (Просмотреть страницу), чтобы увидеть результаты своего труда. Вы должны увидеть нечто, похожее на рис. 5.17.

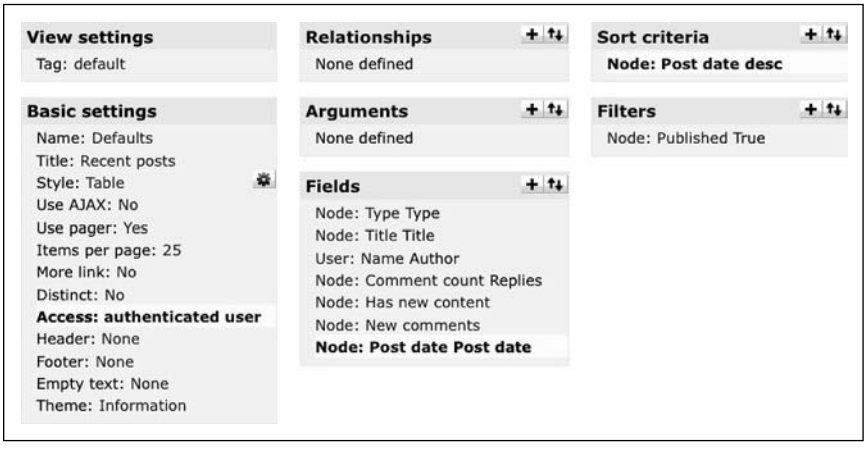


Рис. 5.16. *Настройки представления Recent posts (Последние материалы)*

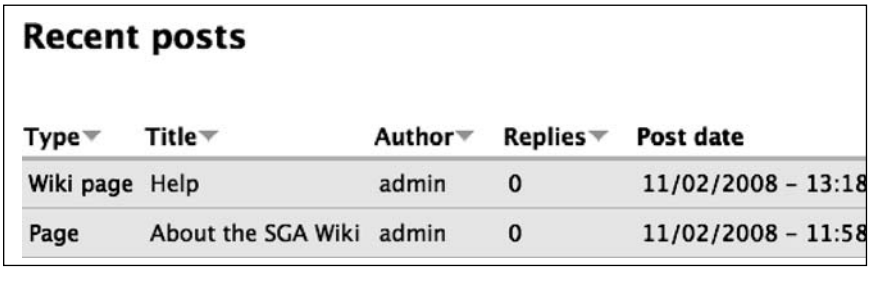


Рис. 5.17. *Представление, отображающее список последних материалов на сайте*

Список Recent edits

Создав представление Recent posts (Последние материалы), мы заканчиваем разработку сайта созданием представления Recent edits (Последние изменения).

1. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Views (*admin/build/views*) и щелкните на ссылке Clone (Скопировать) для представления Recent posts (Последние материалы), которое было создано в предыдущем разделе, так как оно очень похоже на представление Recent edits (Последние изменения).

При создании представления «Recent edits» (Последние изменения) мы использовали инструмент копирования модуля Views, чтобы создать два отдельных представления. Как вариант, можно было бы использовать единственное представление и создать в нем еще одно отображение типа Page (Страница) для списка «Recent edits» (Последние изменения), переопределив в нем необходимые настройки. Оба способа в равной степени приемлемы, и выбор того или другого в основном зависит от того, как организованы ваши представления и степенью схожести содержимого.

2. На начальной странице View settings (Настройки представления) введите значения в соответствии с табл. 5.6 и щелкните на кнопке Next (Далее).

Таблица 5.6. Настройки представления Recent edits (Последние изменения)

View settings (Настройки представления)	Значение
Name (Имя)	tracker_edits
Description (Описание)	Shows all newly edited content (Отображает все измененные материалы)

3. Выделите настройки отображения Defaults (По умолчанию).
4. В группе Basic settings (Базовые настройки) измените значение параметра Title (Заголовок) на «Recent edits» (Последние изменения), чтобы он соответствовал этому новому элементу содержимого.
5. В разделе Fields (Поля) щелкните на ярлыке с изображением знака плюс и включите поле Node: Updated date (Узел: Дата изменения). Дайте этому полю название «Last edited» (Дата последнего изменения).
6. В разделе Sort criteria (Критерий сортировки) удалите существующий критерий сортировки, щелкнув на ссылке Node: Post date (Узел: Дата) и затем на кнопке Remove (Удалить).
7. Щелкните на ярлыке с изображением знака плюс и включите критерий Node: Updated date (Узел: Дата изменения). Установите для параметра Sort order (Порядок сортировки) значение Descending (По убыванию).
8. Измените ссылку меню на эту страницу, чтобы она не конфликтовала со ссылкой в предыдущем представлении. Выберите тип отображения Page (Страница) и в разделе Page settings (Настройки страницы). Установите значения параметров в соответствии с табл. 5.7.

Таблица 5.7. Настройки страницы для представления Recent edits (Последние изменения)

Page: Page settings (Страница: Настройки страницы)	**Значение**
Path (Путь)	tracker-edits
Menu (Меню)	Type (Тип): Normal menu entry (Обычный элемент меню) Title (Заголовок): Recent edits (Последние изменения)

9. По завершении настройки должны выглядеть примерно так, как изображено на рис. 5.18. Завершив настройку параметров, щелкните на кнопке Save (Сохранить).

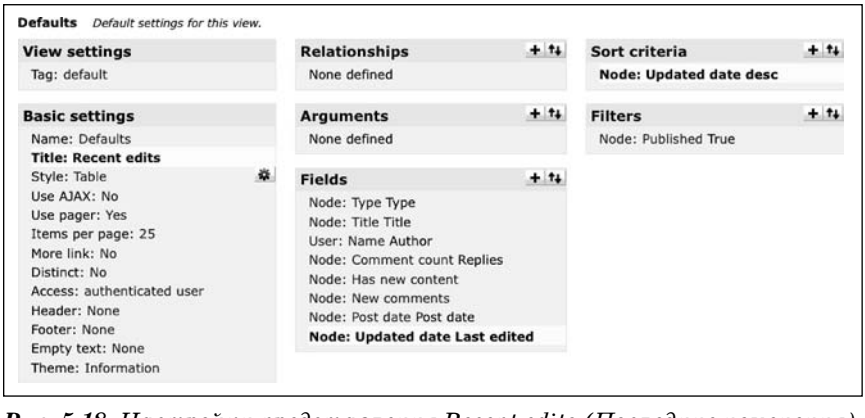


Рис. 5.18. *Настройки представления Recent edits (Последние изменения)*

10. Щелкнув на ссылке View «Page» (Просмотреть страницу), вы должны увидеть результаты, как изображено на рис. 5.19.

Recent edits

Type	Title	Author	Replies	Post date	Last edited
Wiki page	Help	admin	0	11/02/2008 – 13:18	11/02/2008 – 13:30
Page	About the SGA Wiki	admin	0	11/02/2008 – 11:58	11/02/2008 – 12:39

Рис. 5.19. *Представление, отображающее список последних изменений на сайте*

Дополнительно

Мы удовлетворили потребности нашей группы и сумели создать типичный сайт wiki, собрав неплохую коллекцию модулей. Однако существует еще множество модулей, которые можно было бы задействовать для расширения функциональных возможностей. Ниже приводится несколько дополнительных модулей, с помощью которых можно было бы расширить возможности нашего сайта:

PEAR Wiki Filter (http://drupal.org/project/pearwiki_filter)

Вся прелесть PEAR Wiki Filter состоит в том, что благодаря этому модулю вы получаете возможность выбирать из нескольких популярных языков разметки wiki, таких как MediaWiki, TikiWiki и Creole. Недостаток этого модуля заключается в том, что придется немного потрудиться, чтобы установить его. Вам потребуется загрузить отдельные пакеты PHP PEAR для каждого желаемого языка разметки, а также базовый пакет Text Wiki.

Table of Contents (http://drupal.org/project/tableofcontents)

Нередко, когда имеется большой объем справочной информации, могут получаться большие страницы, по которым сложно перемещаться. Модуль Table of Contents отыщет на странице заголовки HTML и построит из них содержание, состоящее из ссылок. Это содержание может быть добавлено в страницу в виде блока.

Talk (http://drupal.org/project/talk)

Модуль Talk перемещает комментарии, оставленные на странице wiki, в отдельную страницу Talk (Обсуждение), имитируя отделение раздела обсуждений, реализованное во многих пакетах wiki.

Представление «Backlinks» для модуля Views (http://drupal.org/project/views)

В состав модуля Views по умолчанию входит представление с именем «Backlinks», которое можно включить на странице управления модулем Views. Представление «Backlinks» создает список всех страниц на сайте, которые ссылаются на страницу, просматриваемую в данный момент.

Taxonomy (базовый)

Еще одно небольшое улучшение, которое можно добавить к сайту, встроено непосредственно в ядро Drupal – модуль Taxonomy. Модуль Taxonomy может помочь сгруппировать содержимое wiki. Например, для ассоциации SGA можно создать словарь «Категории документов» и определить в нем следующие рубрики: «политика», «примечания к встречам» и «планируемые мероприятия». При такой организации члены ассоциации легко смогут получать списки материалов, относящихся к той или иной категории. Еще один иногда полезный способ использования таксономии заключается в создании сло-

варя с произвольными тегами, чтобы пользователи могли добавлять ключевые слова в материалы, создаваемые и редактируемые ими. За дополнительной информацией об использовании таксономии обращайтесь к главе 2.

В заключение

В этой главе было представлено несколько новых модулей: модули Markdown и Freelinking, Pathauto, Token и Diff. Благодаря модулям Markdown и Freelinking мы еще раз рассмотрели некоторые сведения, касающиеся фильтров и форматов ввода в системе Drupal, и увидели, как фактически создается новый формат ввода на основе нескольких сторонних фильтров. Задействовав модули Pathauto и Token, мы поближе познакомились с понятием путей в системе Drupal и узнали, как сделать их более дружественными для людей и поисковых систем. Мы обсудили базовую реализацию механизма редакций в Drupal и рассмотрели, как можно расширить ее возможности с помощью модуля Diff. Напоследок мы воспользовались нашими знаниями модуля Views, полученными в главе 3, чтобы создать несколько представлений, в точности соответствующих потребностям SGA. Мы создали неплохой, надежный сайт wiki, дающий пользователям возможность совместной работы над созданием и редактированием содержимого.

За дополнительной информацией о реализации возможностей среды wiki в Drupal и о том, как эти возможности воплощаются другими, обращайтесь на страницу рабочей группы Wiki, по адресу *http://groups.drupal.org/wiki*.

Ниже приводится перечень модулей, упоминавшихся в этой главе:

- BBCode: *http://drupal.org/project/bbcode*
- Diff: *http://drupal.org/project/diff*
- Freelinking: *http://drupal.org/project/freelinking*
- Liquid Wiki: *http://drupal.org/project/liquid*
- Markdown: *http://drupal.org/project/markdown*
- Path redirect: *http://drupal.org/project/path_redirect*
- Pathauto: *http://drupal.org/project/pathauto*
- PEAR Wiki Filter: *http://drupal.org/project/pearwiki_filter*
- Table of Contents: *http://drupal.org/project/tableofcontents*
- Talk: *http://drupal.org/project/talk*
- Textile: *http://drupal.org/project/textile*
- Token: *http://drupal.org/project/token*
- Views: *http://drupal.org/project/views*

- Wiki: *http://drupal.org/project/wiki*
- Wikitools: *http://drupal.org/project/wikitools*

Ниже приводится список ресурсов, упоминавшихся в этой главе, и ресурсов сообщества для более подробного изучения новых концепций, представленных в этой главе:

- Модули, имеющие отношение к фильтрам: *http://drupal.org/project/Modules/category/63*
- Конвертер Markdown для преобразования текста в разметку HTML: *http://daringfireball.net/projects/markdown/*
- Рабочая группа Wiki: *http://groups.drupal.org/wiki*

6

Управление процессом публикации

Для больших веб-проектов, управляемых содержимым, создание исходной структуры сайта и оформления – это лишь начало работы. Если над созданием информационного наполнения сайта трудится больше, чем несколько человек, то процесс рассмотрения и публикации статей может превратиться в задачу, решить которую под силу будет только Гераклу. Для интернет-газет, интернет-журналов и даже некоторых крупных блогов с большим числом авторов необходимы инструменты, которые позволят редакторам эффективно управлять процессом публикации. В этой главе мы будем использовать встроенный в систему Drupal механизм управления и автоматизации процесса публикации, а также ряд административных модулей, чтобы реализовать «средства управления процессом публикации» для новостного сайта.

В этой главе будут представлены следующие модули:

Trigger (базовый)

Выполняет наиболее типичные задачи при возникновении каких-либо событий на сайте

Workflow (http://drupal.org/project/workflow)

Позволяет администраторам определить собственные этапы рабочего процесса публикации статей, такие как «Рассматриваемый» и «Готов к публикации»

Workspace (http://drupal.org/project/workspace)

Обеспечивает авторам быстрый доступ к их материалам

Views Bulk Operations (http://drupal.org/project/views_bulk_operations)

Позволяет администраторам выполнять типичные действия (такие как публикация или отклонение материала) сразу для нескольких материалов

Если вы собираетесь следовать за практическими упражнениями этой главы, вам следует установить Drupal, используя профиль установки Newspaper, из примеров кода, распространяемых вместе с книгой. В результате применения этого профиля на вашем веб-сервере будет создана заготовка веб-сайта. В окончательном виде этот веб-сайт будет выглядеть, как показано на рис. 6.1 и по адресу: *http://newspaper.usingdrupal.com*. За дополнительной информацией об использовании профилей установки обращайтесь к разделу «Вступление».

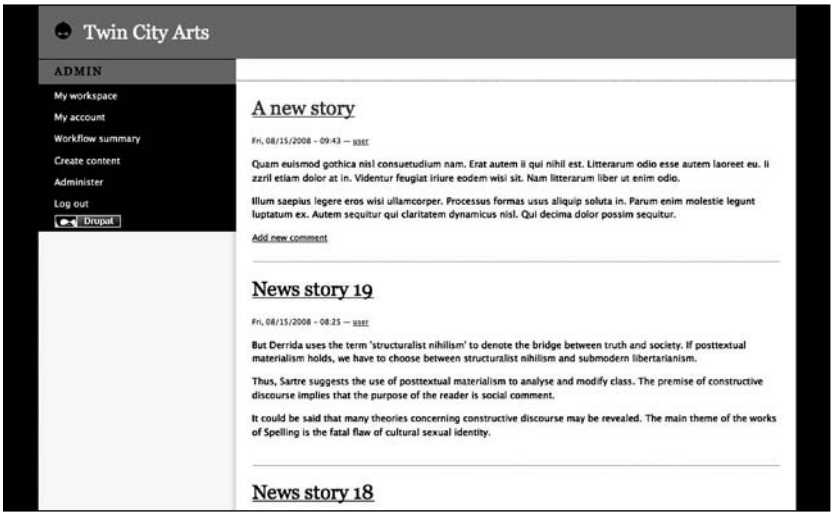


Рис. 6.1. *Вид веб-сайта Twin City Arts*

Задача

Джо и Катрин весь прошлый год занимались созданием новостного сайта для художников, писателей и людей других творческих профессий, проживающих в штате Миннесота: Twin Cities Arts (Город искусств). Он представляет собой комбинацию материалов, поставляемых добровольцами, о событиях в мире художников и писателей и интервью с творческими личностями. Однако по мере роста сайта стало очевидно, что подход к управлению содержимым сайта, основанный на «полном доверии», не оправдал себя. На главной странице стали появляться низкопробные статьи и недостоверные новости, поэтому Джо и Катрин потребовалось как-то упорядочить процесс публикации.

После проведения переговоров с Джо и Катрин и с несколькими их писателями мы получили представление о том, что необходимо для такого сайта. Писатели должны иметь возможность оставлять заметки с новостями, но, прежде чем они станут доступны посетителям, Джо и Катрин хотели бы иметь возможность ознакомиться и одобрить их пу-

бликацию. Когда автор заметки заканчивает свою работу, редакторы должны получить уведомление по электронной почте, чтобы они могли быстро ознакомиться и одобрить заметку для публикации на сайте. Джо и Катрин хотели бы иметь возможность отправлять заметку обратно автору со своими замечаниями и предложениями, при этом автор должен быть уведомлен об этом по электронной почте.

Так как процесс публикации статей состоит из множества стадий, всем, кто вовлечен в него, необходимо иметь возможность ознакомиться со списком заданий, требующих их внимания, для чего необходимо свести этот список в страницу для просмотра. Кроме того, редакторы должны иметь возможность принимать или отвергать сразу несколько статей за один раз, чтобы не терять напрасно времени при работе с большим количеством новостей.

Примечания к реализации

Ядро Drupal позволяет администраторам изменять настройки публикации по умолчанию для каждого типа содержимого, гарантируя, что заметки, отправленные авторами, не попадут на страницы сайта, пока редактор не даст вручную санкции на их публикацию. Однако основная работа, связанная с рассмотрением неопубликованных заметок, их редактированием, публикацией и сохранением изменений, может оказаться неподъемной для сайтов с высокой активностью. Поэтому мы задействуем комплекс модулей, облегчающих этот процесс и обеспечивающих соответствие предъявляемым требованиям.

Процесс публикации

Система Drupal «из коробки» позволяет помечать любые материалы как «опубликовано» или «неопубликовано». Однако для сайта Twin Cities Arts требуется обеспечить более тонкую градацию: необходимо иметь возможность управлять процессом изменения статьи, от момента передачи статьи автором и до момента ее утверждения и публикации. Реализовать это требование можно с помощью модуля Workflow, поэтому мы будем использовать его для создания дополнительных «этапов», которые должна пройти каждая заметка, и управления разрешениями, дающими право передавать заметки с одного этапа на другой. Чтобы позволить редакторам утверждать или отклонять сразу несколько заметок, мы будем использовать модуль Views Bulk Operations.

Извещение по электронной почте

В системе Drupal имеется встроенный механизм отправки сообщений по электронной почте, отображения сообщений на текущей странице и выполнения других действий при возникновении каких-либо событий. Эта особенность, которая называется *действиями* (actions), может объединяться со встроенным модулем Trigger, чтобы обеспечить отправку сообщений редакторам при получении новых заметок.

Некоторые модули, такие как Flag (*http://drupal.org/project/flag*), который обсуждается далее в главе 9, позволяют определять, какие действия должны выполняться при возникновении определенных событий, таких как, например, отметка определенного узла более чем пятью разными пользователями. Модуль Trigger предназначен для решения более общих задач и не обеспечивает подобную возможность тонкой настройки.

Другой модуль, который называется Rules (*http://drupal.org/project/rules*), дублирует функциональные особенности модуля Trigger и добавляет возможность определять условия, которые должны проверяться перед выполнением любых действий. С помощью модуля Rules можно было бы реализовать отправку электронной почты редактору при публикации узлов типа Story (Статья) и не выполнять никаких действий при публикации узлов типа Blog (Блог). Модуль Rules может использоваться для определения весьма сложных условий, но, так как возможностей модуля Trigger достаточно для выполнения наших требований и к тому же он является встроенным модулем, то его использование позволит нам избежать необходимости устанавливать сторонние модули.

Обзорные страницы

Ядро Drupal обеспечивает возможность следить за любыми материалами пользователей на сайте с помощью модуля Tracker. Но, если новые заметки не публикуются автоматически, только пользователи, обладающие разрешением «administer nodes» (управлять узлами), смогут просматривать их. Это означает, что большинство авторов, которые фактически создают информационное наполнение, будут лишены возможности видеть свои собственные статьи или вносить в них изменения после отправки первого, чернового варианта. Модуль Workspace даст каждому пользователю возможность просмотра всех его статей, включая те, которые еще не были опубликованы, на отдельной странице.

Практика: первые шаги

В первую очередь нам необходимо изменить настройки по умолчанию процесса публикации для типа содержимого Story, которые мы будем использовать для наших статей:

1. Перейдите на страницу Administer (Администрирование)→Content management (Содержание)→Content types (Типы содержимого) (*admin/content/nodetype*) и щелкните на ссылке edit (изменить) для типа содержимого Story (Статья).
2. Прокрутите форму с настройками до раздела Workflow settings (Процесс) и разверните его.
3. Снимите флажок Published (Опубликовано).

4. Щелкните на кнопке Save content type (Сохранить тип содержимого), чтобы сохранить изменения.
5. Перейдите на страницу Administer (Администрирование)→User management (Управление пользователями)→Permissions (Разрешения) (*admin/user/permissions*) и выполните настройки так, чтобы все пользователи имели возможность создавать заметки, чтобы редакторы имели полный доступ ко всем материалам и чтобы администраторы сайта могли управлять действиями, о которых мы поговорим ниже в этой главе. Установите разрешения в соответствии с табл. 6.1.

Таблица 6.1. Права доступа к содержимому

Разрешение	**anonymous user (анонимный пользователь)**	**authenticated user (зарегистрированный пользователь)**	**editor (редактор)**	**site administrator (администратор сайта)**
модуль node				
administer nodes (управлять материалами)			отмечен	отмечен
create story content (создавать материал типа story)		отмечен		
edit own story content (изменять собственный материал типа story)		отмечен		
модуль system				
access administration pages (доступ к страницам администрирования)			отмечен	отмечен
administer actions (управлять действиями)				отмечен

6. Щелкните на кнопке Save permissions (Сохранить права доступа).

Теперь, если вы зарегистрируетесь как пользователь *user*, вы сможете отправлять свои заметки, но они не будут появляться на главной странице и «обычные пользователи» их не увидят. Зарегистрировавшись как пользователь *editor* и перейдя на страницу Administer (Администрирование)→Content management (Содержание)→Content (Содержимое) (*admin/content/node*), вы увидите эти статьи, сможете читать и публиковать их.

В центре внимания: действия и триггеры

С имеющимися настройками авторы могут отправлять свои заметки в любой момент, но они останутся неопубликованными, пока Джо или Катрин не опубликуют их вручную. Однако здесь возникает новая проблема: как редакторы узнают, что появилась новая статья, требующая их внимания? В настоящий момент они вынуждены проверять сайт каждые несколько часов, чтобы проверить, не осталось ли что-нибудь без их внимания.

Именно в таких ситуациях на сцену выходит одна из наиболее мощных, но малоизвестных особенностей системы Drupal: *действия*. На языке Drupal действие – это одиночная, легко встраиваемая операция, такая как публикация узла, отображение сообщения текущему зарегистрированному пользователю или отправка подготовленного заранее сообщения по электронной почте на определенный адрес. Сторонние модули легко позволяют организовать выполнение подобных действий при наступлении определенных условий или определенного события.

Насколько это полезно? Основная задача многих модулей заключается в предоставлении пользователям удобных инструментов управления: например, снимать с публикации материал щелчком мыши на ссылке под узлом или автоматически блокировать пользователей, если они посылают сообщения оскорбительного характера. Но что, если вы бы предпочли, чтобы сайт отправлял администратору сообщение по электронной почте при получении оскорбительного сообщения вместо автоматической блокировки пользователя? Что, если вы бы предпочли, чтобы щелчок на ссылке под узлом вызывал бы перемещение материала на главную страницу, а не снятие его с публикации? Вместо того чтобы жестко задавать определенное поведение, разработчики модулей могут предложить администраторам выбирать требуемое поведение из списка доступных действий, как показано на рис. 6.2, и затем настраивать его необходимым образом. Это предоставляет администраторам и другим создателям сайтов гораздо большую гибкость, при этом не требуя от разработчиков модулей предусматривать и реализовывать все возможные действия.

Существует два типа действий: базовые действия, которые выполняют простейшие операции, и сложные действия, которые требуют дополнительной настройки. Многие административные задачи в Drupal могут выполняться с помощью действий. Блокирование пользователей, перемещение материалов на главную страницу, добавление рубрик таксономии в содержимое, удаление комментариев, содержащих оскорбительные выражения, отправка электронной почты и тому подобное – все это вполне возможно. Кроме того, сторонние модули могут добавлять новые действия. Примерами таких полезных действий могут служить: «Отправить текстовое сообщение по указанному номеру телефона» или «Наделить текущего зарегистрированного пользователя новой ролью». Некоторые действия, такие как «Отправить электронную почту», име-

ют дополнительные настройки. Имеется возможность создавать множество копий действия, каждая из которых будет иметь собственные настройки. Например, одна копия может отправлять администратору по электронной почте извещение о том, что была создана новая учетная запись, а вторая – извещать автора статьи о появлении нового комментария.

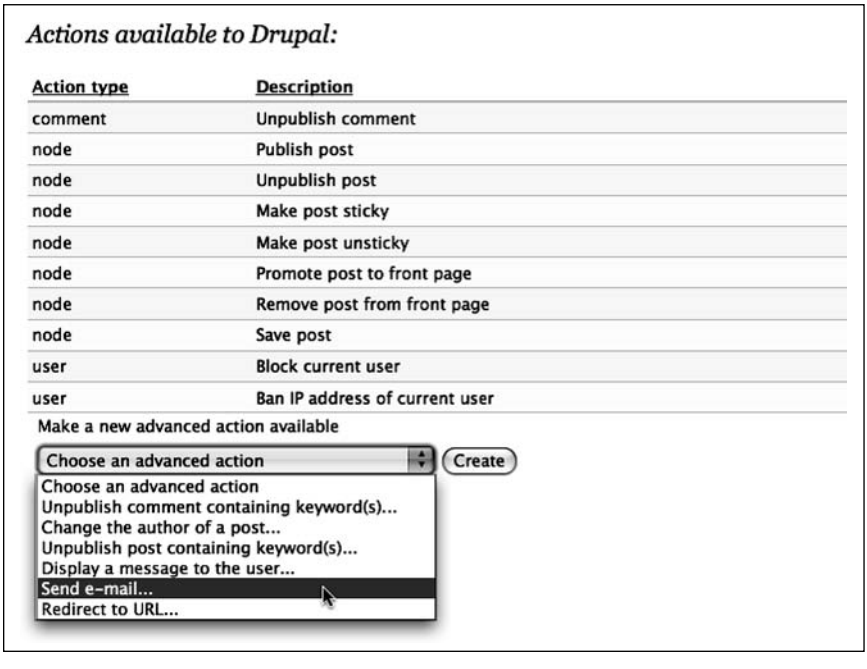


Рис. 6.2. *Действия, доступные в Drupal по умолчанию*

В состав ядра Drupal входит простой модуль с названием Trigger, страница с настройками которого показана на рис. 6.3. Этот модуль позволяет администраторам связывать действия с различными событиями, возникающими на сайте. Например, триггер «User account created» (Создана учетная запись) может быть связан с действием «Send email» (Отправить сообщение по электронной почте), чтобы автоматически извещать администраторов при появлении новых пользователей. Модуль Trigger имеет встроенную поддержку основных событий, имеющих отношение к содержимому, таких как создание, изменение или удаление узлов и комментариев. Кроме того, он поддерживает события, связанные с учетными записями пользователей и рубриками таксономии, – если вы пожелаете получать извещения, когда блоггеры на вашем сайте будут добавлять новые произвольные рубрики в свои статьи, то это вполне возможно! Сторонние модули могут также определять свои собственные события, доступные модулю Trigger, обеспечивая единый и непротиворечивый интерфейс связывания действий с важными событиями.

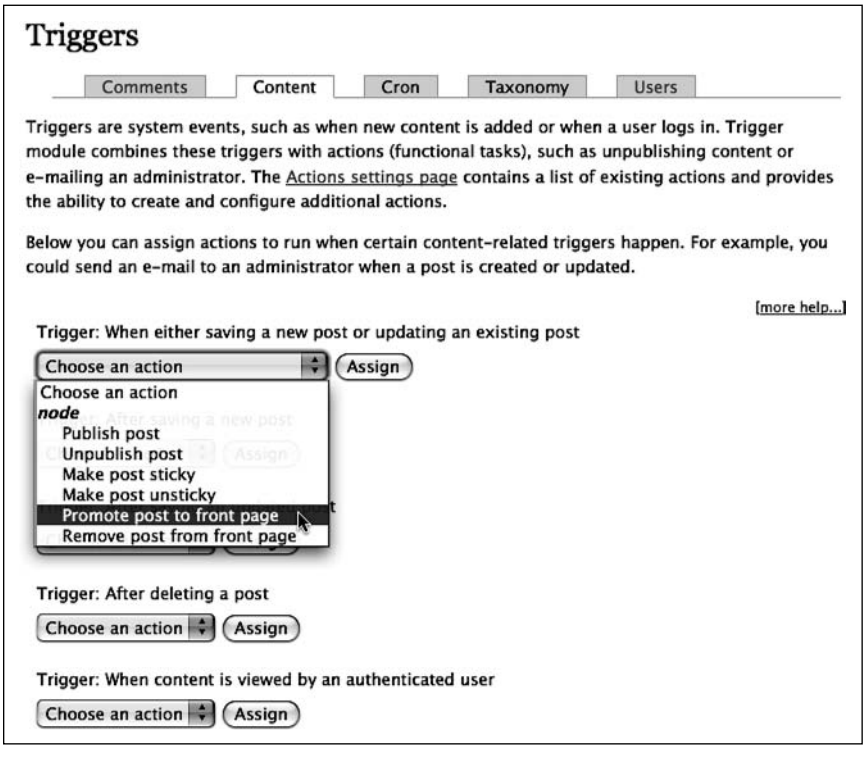


Рис. 6.3. *Модуль Trigger позволяет связывать действия с системными событиями*

Что касается веб-сайта Twin City Arts, мы будем использовать действия и модуль Trigger, чтобы извещать редакторов сайта о появлении новых материалов, требующих их внимания, а также чтобы извещать авторов, когда статьи будут признаны не готовыми для публикации и возвращены им на доработку.

Практика: действия и триггеры

Как уже упоминалось ранее, некоторые действия требуют дополнительной настройки, прежде чем их можно будет связать с определенными событиями. Мы должны определить текст сообщения, отправляемого по электронной почте тому, кто оставил свою заметку на сайте, а также текст сообщения, которое авторы будут видеть после отправки своих статей, как это показано на рис. 6.4.

Настройка действий

1. Перейдите на страницу Administer (Администрирование)→Site configuration (Настройка сайта)→Actions (Действия) (*admin/settings/*

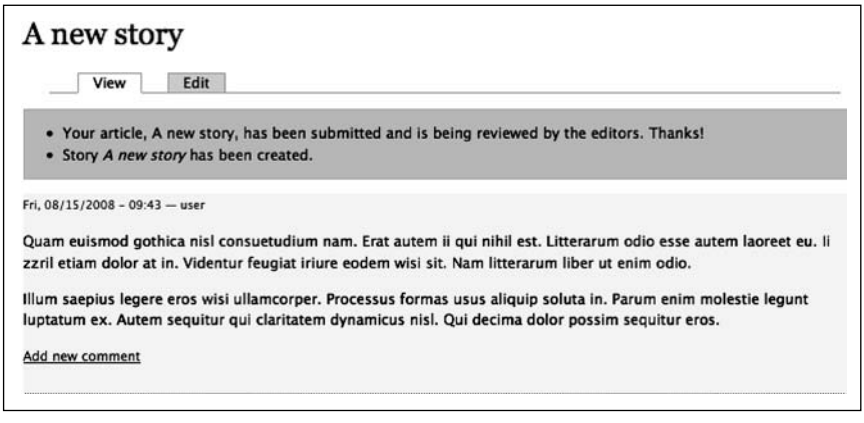


Рис. 6.4. *Сообщение, отображаемое перед автором*

actions) и щелкните на раскрывающемся списке под заголовком Make a new advanced action available (Новое сложное действие), как было показано на рис. 6.2. Вы увидите перечень доступных действий. Выберите действие Send email... (Отправить сообщение электронной почты...) и щелкните на кнопке Create (Создать).

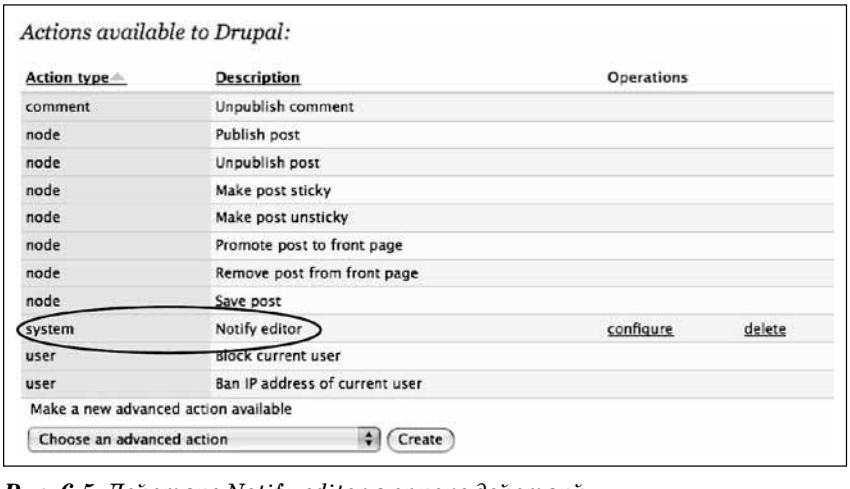
2. Заполните поля формы в соответствии с табл. 6.2.

Таблица 6.2. Настройки действия отправки сообщения редактору

Поле	Значение
Description (Описание)	Notify editor (Известить редактора)
Recipient (Получатель)	[ваш адрес электронной почты]
Subject (Тема)	New article submitted by %username (Получена новая статья от пользователя %username)
Message (Сообщение)	%username submitted a new %node_type titled «%title» for approval at %site_name. --- %teaser --- To review the full article and publish or delete it, visit %node_url. (Пользователь %username создал новый материал типа %node_type с названием «%title» для публикации на сайте %site_name.---%teaser---Чтобы ознакомиться с полным текстом статьи и опубликовать или удалить ее, перейдите по ссылке %node_url.)

Обратите особое внимание на специальные заполнители, использованные в тексте сообщений. Всякий раз, когда вызывается некоторое действие, заполнители %username, %title и %node_url автоматически замещаются информацией, извлеченной из созданного узла. Они напоминают лексемы, используемые в модуле Token, о котором рассказывалось в предыдущей главе. Полный список доступных заполнителей приводится в справочном тексте под текстовой областью Message (Сообщение).

3. Щелкните на кнопке Save, после чего вы вернетесь на главную страницу Actions (Действия), где увидите наше новое действие с типом system (системное) в колонке Action type (Тип действия), как изображено на рис. 6.5.



Actions available to Drupal:

Action type	Description	Operations	
comment	Unpublish comment		
node	Publish post		
node	Unpublish post		
node	Make post sticky		
node	Make post unsticky		
node	Promote post to front page		
node	Remove post from front page		
node	Save post		
system	Notify editor	configure	delete
user	Block current user		
user	Ban IP address of current user		

Make a new advanced action available

Choose an advanced action | Create

Рис. 6.5. *Действие Notify editor в списке действий*

4. Нам также надо показать сообщение автору статьи, чтобы он знал, что его статья не исчезла в никуда. Для этого мы настроим новое действие. Выберите действие Display a message to the user... (Показать сообщение пользователю...) из списка Make a new advanced action available (Новое сложное действие) и щелкните на кнопке Create (Создать).
5. Заполните поля формы в соответствии с табл. 6.3.

Таблица 6.3. Настройки действия отображения сообщения автору

Поле	Значение
Description (Описание)	Placate author (Успокоить автора)
Message (Сообщение)	% title has been submitted and is being reviewed by the editors. Thanks! (Статья % title была принята и передана на рассмотрение редакторам. Спасибо!)

6. По окончании щелкните на кнопке Save (Сохранить).
7. Последнее действие, которое нам необходимо, – известить автора, что его статья возвращена на доработку. Как и в случае отправки сообщения редактору, выберите из списка Make a new advanced action available (Новое сложное действие) действие Send e-mail... (Отправить сообщение электронной почты...) и щелкните на кнопке Create (Создать).
8. Заполните поля формы в соответствии с табл. 6.4 и щелкните на кнопке Save (Сохранить).

Таблица 6.4. Настройки действия отправки сообщения автору

Поле	Значение
Description (Описание)	Agitate author (Озадачить автора)
Recipient (Получатель)	% author
Subject (Тема)	% title needs more work (Статья % title требует доработки)
Message (Сообщение)	Hello % username,The editors have reviewed your article, % title, submitted to % site_name, and have determined that it needs a bit more work before publication. Feel free to resubmit the article for another review. (Здравствуйте, % username.Редакторы просмотрели вашу статью % title, отправленную для публикации на сайте % site_name, и определили, что она требует некоторой доработки, прежде чем ее можно будет опубликовать. Предлагаем вам повторно выслать вашу статью после доработки.)

Назначение триггеров

Теперь, когда мы настроили необходимые нам действия, можно сделать следующий шаг и задействовать модуль Trigger, чтобы связать определенные события с действиями, созданными нами.

1. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Modules (Модули) (*admin/build/modules*) и включите следующие модули:
   - Core – optional (Ядро – дополнительные)
     - Trigger
2. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Triggers (Триггеры) (*admin/build/trigger/node*). Вы окажетесь на вкладке Content (Содержимое), на странице управления модулем Trigger.

3. В разделе Trigger: After saving a new post (Триггер: После сохранения нового сообщения) выберите действие Notify editor (Известить редактора) и щелкните на кнопке Assign (Назначить). Этот раскрывающийся список изображен на рис. 6.6.
4. В том же разделе Trigger: After saving a new post (Триггер: После сохранения нового сообщения) выполните те же манипуляции для действия Placate author (Озадачить автора). Щелкните на кнопке Assign (Назначить), чтобы добавить его в наш список.

Это все, что необходимо для запуска наших действий. Теперь всякий раз, когда пользователи будут добавлять новые материалы, они будут оставаться неопубликованными, но при этом редакторам будут отправляться извещения с прямыми ссылками на статьи, чтобы они могли ознакомиться с ними и решить – возможно ли их опубликовать. Кроме того, мы сообщаем авторам статей, что их материалы не пропали, чтобы они не впадали в недоумение, заметив, что их статьи не появляются на сайте немедленно.

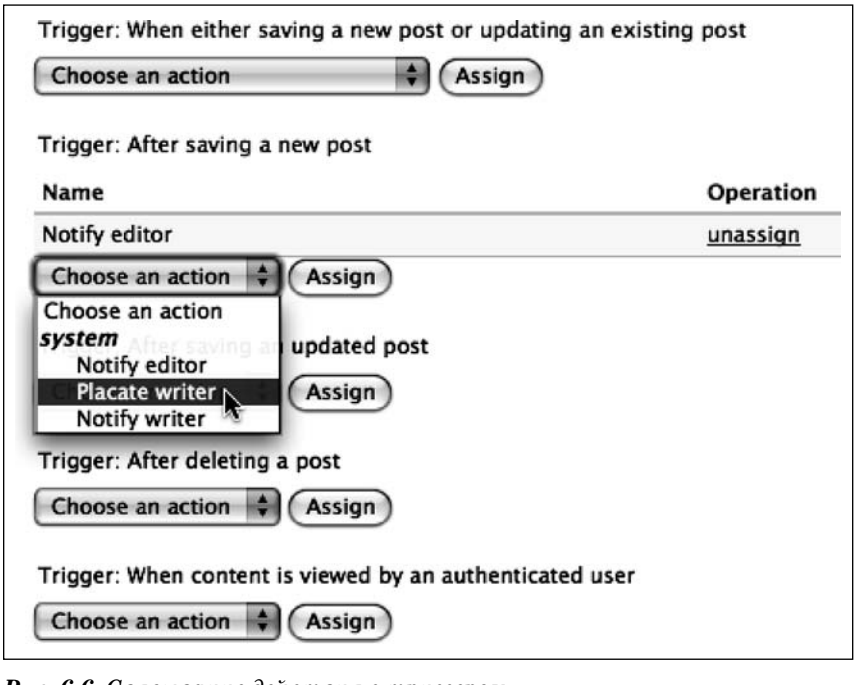


Рис. 6.6. *Связывание действия с триггером*

В центре внимания: модуль Workflow

До сих пор мы полагались на встроенный флаг published «опубликовано», который сопровождает каждый материал в системе Drupal. Когда он снят, просматривать содержимое могут только пользователи, об-

ладающие разрешением «administer nodes» (управлять материалами). Для некоторых сайтов этого вполне достаточно, но он не обеспечивает полный контроль над материалами, необходимый редакторам и авторам. Например, автор не имеет возможности пометить свою заметку, как находящуюся в процессе написания, – с тем, чтобы вернуться к работе над ней позже. Кроме того, редактор не имеет простой возможности сообщить автору, что статья требует доработки – редактор должен явно сообщить об этом.

Именно эту проблему призван решить модуль Workflow (*http://drupal.org/project/workflow*). Он позволяет администраторам сайта определять этапы, называемые состояниями, которые должен пройти каждый материал, прежде чем он будет опубликован. На комплексных сайтах со строгими юридическими ограничениями может потребоваться, чтобы материалы проходили этапы «Editorial review» (Ожидает рассмотрения редактора), «Legal review» (Ожидает рассмотрения юриста), «Executive approval» (Ожидает утверждения к публикации) и «Ready to publish» (Готово к публикации). Когда материал находится на одном из этих этапов, только пользователи с определенными ролями могут передавать его на следующий этап. Тем самым гарантируется, что только наделенные соответствующими полномочиями лица смогут одобрить материал, прежде чем он попадет на страницы сайта.

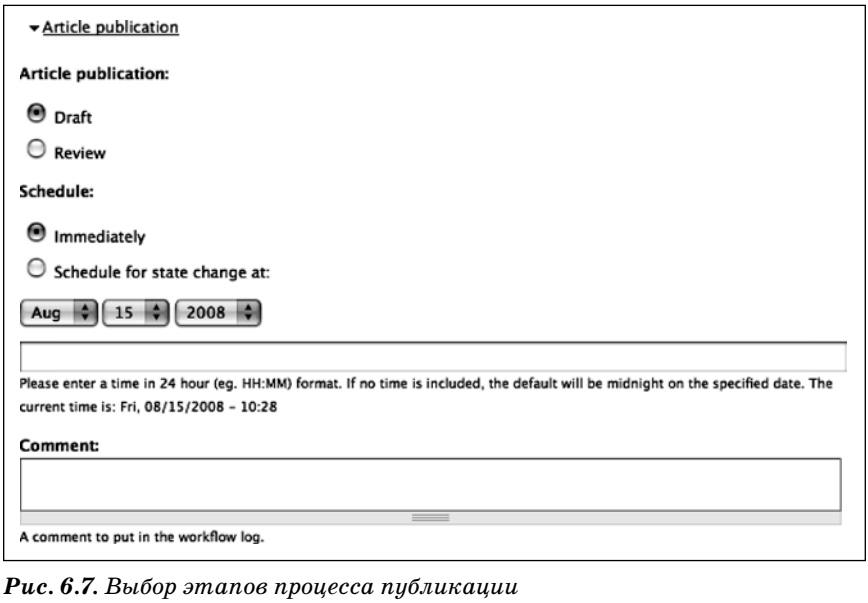


Рис. 6.7. *Выбор этапов процесса публикации*

Выбор этапов процесса публикации может быть выполнен непосредственно при создании или редактировании узла посредством новой группы параметров, добавленной в форму узла, как показано на рис. 6.7. Наряду с выбором этапа пользователь может также оставить свой ком-

ментарий в журнале процесса публикации. Благодаря этому другие смогут увидеть описание причин или примечания к изменениям. Кроме того, модуль Workflow дает пользователям возможность привязать изменение этапа к определенному времени. Например, переход с этапа «Executive approval» (Ожидает утверждения к публикации) на этап «Ready to publish» (Готово к публикации) может быть запланирован на 8:00 часов утра, даже если вице-президент принял решение в 11:00 вечера предыдущего дня.

В дополнение к элементам управления формы узла модуль Workflow предоставляет вкладку, следующую за вкладками View (Просмотреть) и Edit (Изменить). Эта вкладка отображает этап и элементы управления планированием, а также журнал хронологии процесса (Workflow History), в котором отслеживаются переходы с одного этапа на другой и отображаются оставленные комментарии. Пример такого журнала показан на рис. 6.8.

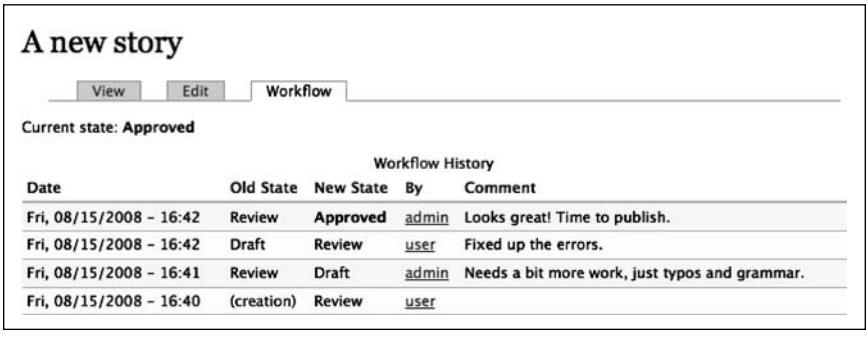


Рис. 6.8. *Журнал хронологии процесса публикации*

Что еще более важно, модуль Workflow может использовать действия Drupal. Каждый раз, когда материал переходит с одного этапа на другой, может вызываться определенное действие. Например, когда редактор переносит материал с этапа «Editorial review» (Ожидает рассмотрения редактора) на этап «Legal review» (Ожидает рассмотрения юриста), имеется возможность отправить в юридический отдел извещение по электронной почте. Кода вице-президент подписывает материал к публикации, перемещая его с этапа «Executive approval» (Ожидает утверждения к публикации) на этап «Ready to publish» (Готово к публикации), с помощью действий можно автоматически опубликовать материал и поместить его на главную страницу.

Эта комбинация инструментов (действия и модули Trigger и Workflow) позволяет использовать на сайтах сложные процессы подготовки материалов к публикации и замысловатые механизмы утверждения. Более того, для каждого типа содержимого может быть предусмотрен свой процесс подготовки. Для сообщений в блогах можно предусмотреть упрощенный процесс, а для официальных материалов, таких как стра-

ницы «About us» (О компании) и «Privacy policy» (Политика конфиденциальности), – более строгую систему утверждения.

В дополнение к модулю Workflow в пакет Workflow входит модуль Workflow Access. Он позволяет скрывать материалы от пользователей в зависимости от их ролей и текущего этапа, на котором находится материал. В нашем примере только пользователи с ролью «legal team» (юридический отдел) могли бы получить доступ к материалам, находящимся на этапе «Legal review» (Ожидает рассмотрения юриста).

Практика: определение процесса публикации

Для сайта Twin City Arts нам необходимо определить несколько этапов, через которые должны будут проходить заметки. Нам потребуется этап «Draft» (Рукопись), чтобы авторы могли сохранять промежуточные версии, еще не готовые для рассмотрения редактором; также нам потребуется этап «Review» (На рассмотрении), который начинается, как только автор передаст заметку на рассмотрение. Наконец, нам потребуется этап «Approved» (Одобрено), на котором заметка публикуется на сайте и становится доступна для всеобщего обозрения.

1. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Modules (Модули) (*admin/build/modules*) и включите следующий модуль:
   - Workflow
     - Workflow
2. Затем перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Workflow (*admin/build/workflow*). Вы увидите сообщение: «No workflows have been added. Would you like to add a workflow?» (Ни одного процесса еще не было добавлено. Желаете добавить процесс?). Чтобы создать определение процесса публикации, щелкните на вкладке Add workflow (Добавить процесс) (*admin/build/workflow/add*).
3. Введите в поле Workflow Name (Название процесса) текст «Article publication» (Публикация статьи) и щелкните на кнопке Add workflow (Добавить процесс).
4. После этого будет выполнен переход на страницу со списком процессов, изображенную на рис. 6.9, где можно будет увидеть наш новый процесс «Article publication» (Публикация статьи). В нижней части страницы присутствует список типов содержимого. Нам требуется применить наш процесс к типу содержимого Story (Статья), так как именно этот тип будет использоваться для статей. Щелкните на раскрывающемся списке в колонке Workflow (Процесс) для типа содержимого Story (Статья) и выберите пункт Article publication (Публика-

ция статьи). Щелкните на кнопке Save workflow mapping (Сохранить схему привязки процессов), чтобы сохранить изменения.

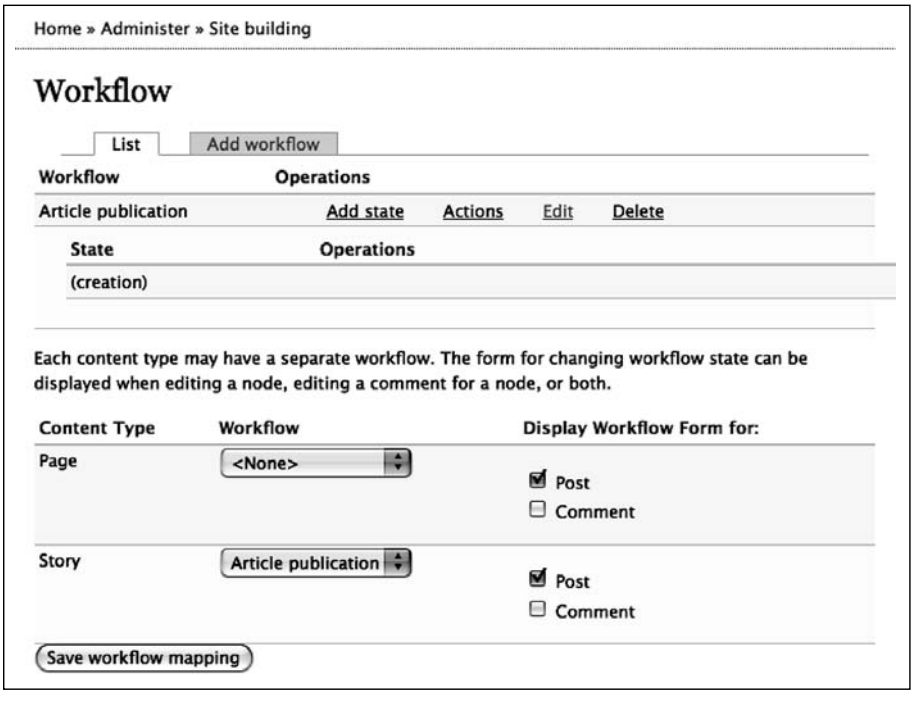


Рис. 6.9. *Настройки процесса публикации статей*

5. Мы привязали тип содержимого Story (Статья) к процессу публикации, но этот процесс пока ничего еще не делает. Нам необходимо добавить в него этапы, которые должны будут проходить статьи. Щелкните на ссылке Add state (Добавить этап). Введите текст «Draft» (Рукопись) в поле State name (Название этапа) и щелкните на кнопке Save (Сохранить).
6. Выполните те же действия, щелкнув на ссылке Add state (Добавить этап), и добавьте два других этапа: «Review» (На рассмотрении) и «Approved» (Одобрено). По окончании список этапов процесса должен выглядеть, как показано на рис. 6.10.
7. Определите, кто может производить перемещение статьи с этапа на этап, отметив роли в соответствии с табл. 6.5. Обратите внимание: мы преднамеренно не указываем роли для этапа «Approved» (Одобрено), потому что, как только статья будет одобрена и опубликована, она не может «вернуться» обратно на этап «Draft» (Рукопись) или «Review» (На рассмотрении).

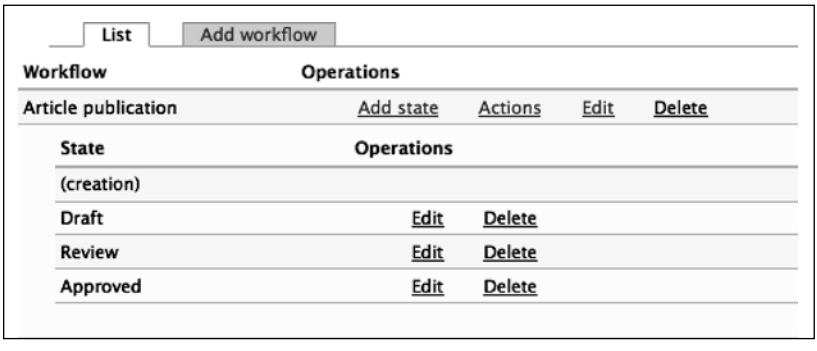


Рис. 6.10. *Список этапов процесса Article publication (Публикация статьи)*

Таблица 6.5. Настройки порядка перемещения статей для процесса «Article publication»

Откуда/Куда	**Draft (Рукопись)**	**Review (На рассмотрении)**	**Approved (Одобрено)**
(creation) (создание)	author	author	
Draft (Рукопись)		author	
Review (На рассмотрении)	editor		editor
Approved (Одобрено)			

8. Последнее, что нам требуется сделать на данной странице, – это определить, кому будет доступна вкладка Workflow (Процесс) при просмотре узлов. Обычный набор параметров настройки процесса публикации будет отображаться для всех непосредственно в форме узла. Те, кому будет доступна вкладка, получат возможность просматривать хронологию процесса. Мы сделаем вкладку доступной для наших редакторов, отметив роль editor (редактор) в настройках разрешений для вкладки Workflow (Процесс). По окончании щелкните на кнопке Save (Сохранить).
9. Наконец, нам осталось еще кое-что: действие, отправляющее редакторам извещение по электронной почте, запускается всякий раз, когда выполняется сохранение узла. Но мы хотим оповещать редакторов, только когда статья передается на рассмотрение, а не когда она создается. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Workflow (Процесс) (admin/build/workflow) и щелкните на ссылке Actions (Действие) для процесса Article publishing (Публикация статьи). В результате вы снова пере-

меститесь на страницу настройки модуля Trigger, но на этот раз окажетесь на вкладке Workflow (Процесс).

10. Вам будет предложен список переходов между этапами, который мы уже определили для нашего процесса. Назначьте действия в соответствии с табл. 6.6, щелкая на кнопке Assign (Назначить) для каждого действия, добавляемого в триггер. По окончании страница с настройками должна выглядеть, как показано на рис. 6.11. Кроме того, будет неплохо удалить действия, назначенные на странице Triggers (Триггеры) (*admin/build/trigger/node*). Теперь, когда действия инициируются при изменении этапа процесса, они больше не нужны.

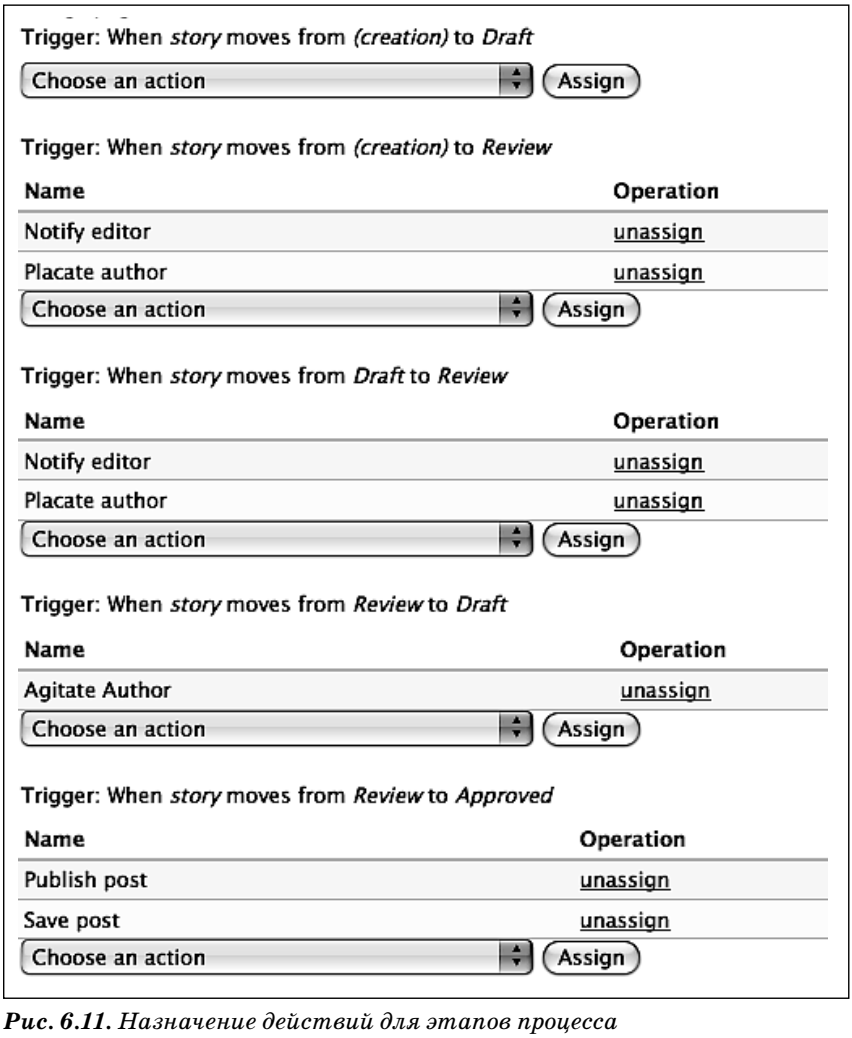


Рис. 6.11. *Назначение действий для этапов процесса*

Таблица 6.6. Назначение действий триггерам процесса

Триггер	Значение
When story moves from (creation) to Review (Когда заметка перемещается с этапа (creation) на этап Review)	system (системный): Notify editor (Известить редактора) system (системный): Placate author (Успокоить автора)
When story moves from Draft to Review (Когда заметка перемещается с этапа Draft на этап Review)	system (системный): Notify editor (Известить редактора) system (системный): Placate author (Успокоить автора)
When story moves from Review to Draft (Когда заметка перемещается с этапа Review на этап Draft)	system (системный): Agitate author (Озадачить автора)
When story moves from Review to Approved (Когда заметка перемещается с этапа Review на этап Approved)	node (узел): Publish post (Опубликовать)

При добавлении действия Publish post (Опубликовать) Drupal автоматически добавит действие Save post (Сохранить сообщение) и отобразит сообщение, объясняющее причину: «You have added an action that changes the property of a post. A Save post action has been added so that the property change will be saved» (Вы добавили действие, изменяющее свойство отправления. Действие Save post (Сохранить сообщение) было добавлено с целью сохранить измененное свойство.). Так как при выполнении этого действия изменяется состояние узла, это изменение необходимо сохранить.

11. Если теперь перейти на страницу Create content (Создать материал)→Story (Статья) (*node/add/story*) и создать новый материал с типом Story, в группе параметров Article publication (Публикация статьи) можно увидеть, что автор имеет возможность выбрать либо этап Draft (Рукопись), либо Review (На рассмотрении), но не Approved (Одобрено), как показано на рис. 6.12.

Двинемся дальше и создадим еще одну-две заметки. Переместите хотя бы одну заметку на этап Review (На рассмотрении), в результате вы должны увидеть сообщение, извещающее, что ваша заметка передана на рассмотрение.

Если теперь зарегистрироваться на сайте с правами администратора и посетить вкладку Workflow (Процесс) на странице с заметкой, находящейся на этапе Review (На рассмотрении), можно заметить, что появилась дополнительная возможность переместить заметку на этап Approved (Одобрено), недоступная обычным пользователям.

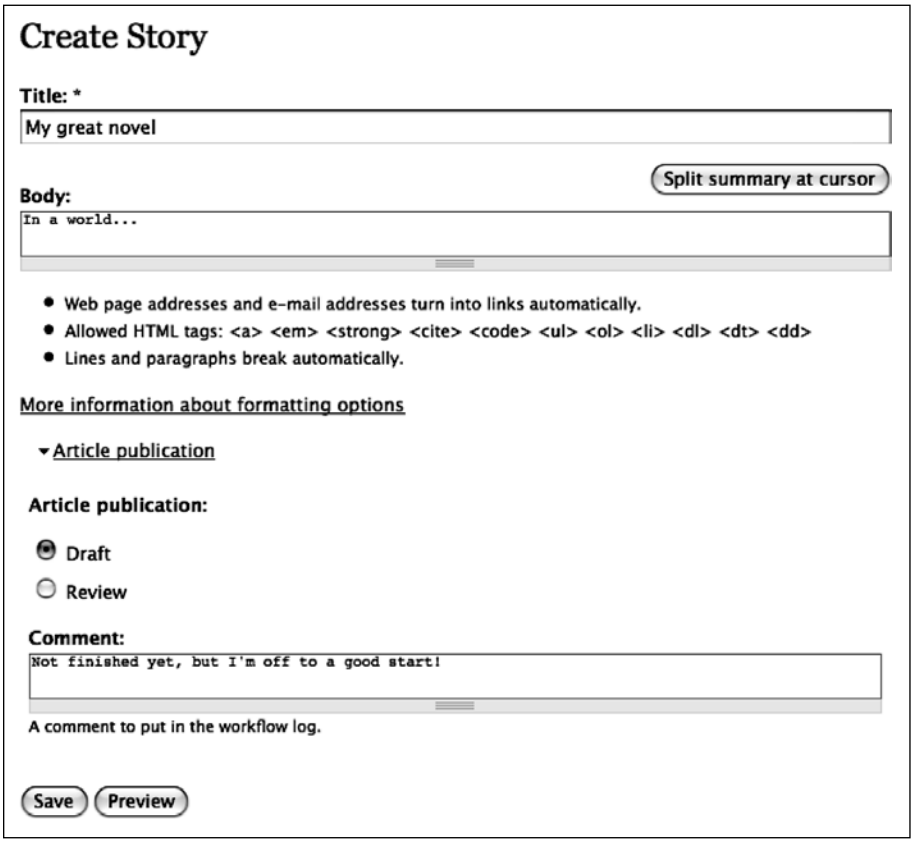


Рис. 6.12. *Процесс публикации статьи в действии*

В центре внимания: модуль Workspace

Мы практически реализовали функциональность, необходимую Джо и Катрин, но остается нерешенной еще одна заметная проблема. Авторы могут создавать новые заметки и оставлять их на этапе Draft (Рукопись), если они еще не готовы к передаче на рассмотрение. Что случится, когда автор пожелает вернуться к заметке и закончить работу? Так как такая заметка еще не опубликована, система Drupal не включит ее ни в один из обычных списков материалов! Автор сможет вернуться к рукописи, только если он запомнит идентификационный номер своей статьи, а для этого нужно быть немного больше, чем добровольцем.

Мы могли бы с помощью модуля Views создать собственную страницу со списком, включающим все материалы, даже неопубликованные, принадлежащие текущему пользователю. Однако то же самое можно реализовать с помощью модуля Workspace (*http://drupal.org/project/*

workspace), причем намного быстрее и проще. Этот модуль создает для каждого пользователя его собственную страницу со списком материалов, комментариев и файлов, отправленных им на сайт, как показано на рис. 6.13. По умолчанию модуль Workspace добавляет новый пункт меню My workspace (Мое рабочее пространство), обеспечивающий доступ к этой странице. Однако вы можете также добавить ссылку на страницу рабочего пространства непосредственно на персональную страницу каждого пользователя.

Помимо страницы, напоминающей представление, модуль Workspace также обеспечивает возможность быстрого создания нового содержимого. Список типов содержимого, создание которых доступно пользователю, отображается в виде раскрывающегося списка на его странице рабочего пространства вместе с удобной кнопкой Add new item (Создать новый элемент), щелчок на которой немедленно открывает страницу создания содержимого, такую как Create content (Создать материал)→Story (Статья) (*node/add/story*). Это делает страницу рабочего пространства простой универсальной панелью инструментов для постоянных авторов.

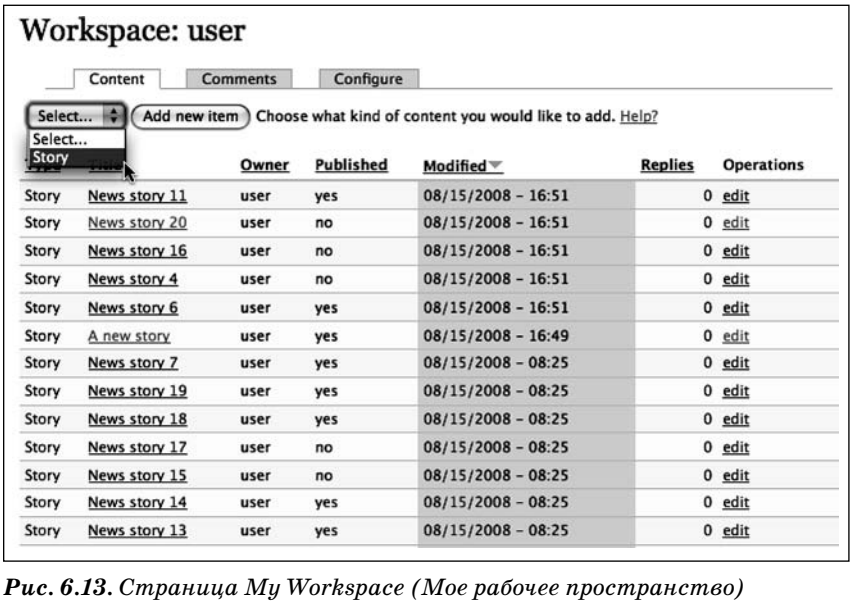


Рис. 6.13. *Страница My Workspace (Мое рабочее пространство) со списком материалов пользователя*

Практика: создание рабочих пространств

Модуль Workspace чрезвычайно прост в настройке. Когда мы закончим выполнение нескольких простых шагов, все наши пользователи получат страницу, изображенную на рис. 6.13.

1. Для начала перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Modules (Модули) (*admin/build/modules*) и включите следующий модуль:
   - Other (Другой)
     - Workspace
2. Вот и все! Теперь вы должны увидеть ссылку My workspace (Мое рабочее пространство) в навигационном меню, и каждый пользователь на сайте получает свое собственное рабочее пространство.
3. Теперь на странице Administer (Администрирование)→User management (Управление пользователями)→Permissions (Разрешения) (*admin/user/permissions*) вы увидите, что модуль Workspace также определяет некоторые разрешения. Если пользователям дать разрешение «administer own workspace» (управлять своим рабочим пространством), они смогут определять такие параметры, как количество элементов, отображаемых в списке. Определите разрешения в соответствии с табл. 6.7 и щелкните на кнопке Save permissions (Сохранить права доступа).

Таблица 6.7. Настройка разрешений для модуля Workspace

Разрешение	**anonymous user (анонимный пользователь)**	**authenticated user (зарегистрированный пользователь)**	**editor (редактор)**	**site administrator (администратор сайта)**
administer own workspace (управлять собственным рабочим пространством)		отмечен	отмечен	отмечен
administer workspaces (управлять рабочими пространствами)				отмечен
view all workspaces (просматривать все рабочие пространства)				отмечен

В центре внимания: модуль Views Bulk Operations

Мы пока не слишком позаботились о редакторах нашего веб-сайта. Сейчас они могут перемещать статьи с одного этапа на другой, возвращать их авторам на доработку, получать извещения при передаче статей на

рассмотрение и автоматически публиковать статьи, когда они будут готовы. Однако редакторы все еще вынуждены работать со статьями по отдельности, что является весьма неприятным ограничением, особенно когда сайт переполняется заметками о незначительных новостях или плохими стихами.

В системе Drupal имеется встроенная страница управления содержимым Administer (Администрирование)→Content management (Содержание)→ Content (Содержимое) (*admin/content/node*). Она выводит список, содержащий до 20 материалов, и позволяет выполнять массовые операции над ними, такие как публикация или снятие с публикации, как показано на рис. 6.14. Однако эта страница не использует преимущества системы действий и не позволяет определять, какая информация должна отображаться в таблице.

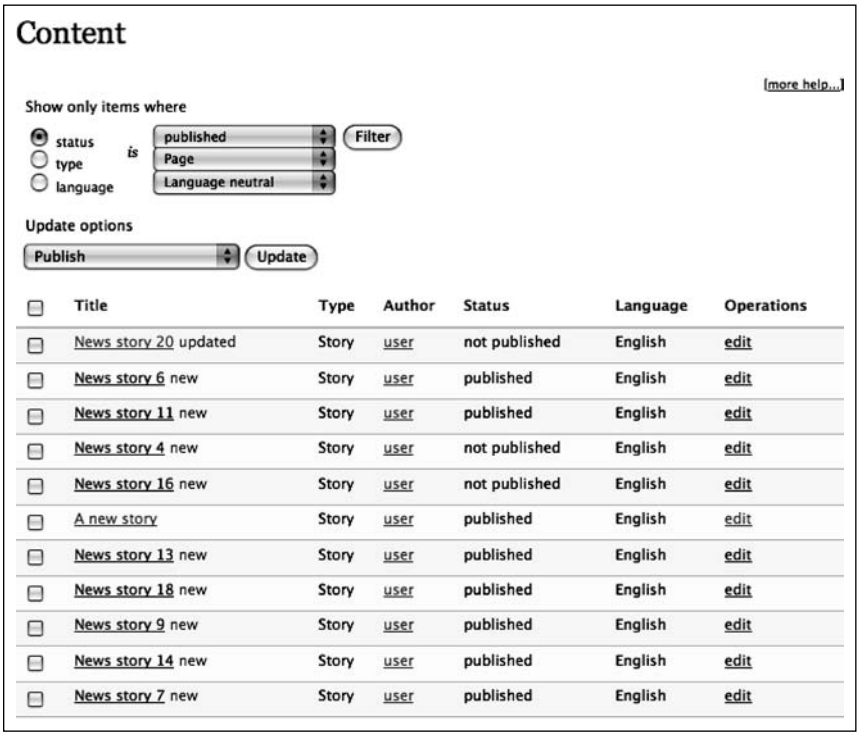


Рис. 6.14. *Базовая страница Drupal управления материалами*

Модуль Views Bulk Operations (*http://drupal.org/project/views_bulk_operations*) разрабатывался специально для преодоления этих ограничений. Он содержит новое представление с именем «Bulk Operations» (Массовые операции). Это представление отображает содержимое почти так же, как стандартное табличное представление, – со строками, колонками и заголовками. Однако в нем для каждого узла присутству-

ет флажок, а в верхней части страница имеется раскрывающийся список, позволяющий выбирать действие. В результате получается простая в настройке форма управления, обладающая всеми возможностями стандартной страницы управления материалами, но обладающая гораздо большей гибкостью. Пример представления для выполнения массовых операций показан на рис. 6.15.

Кроме того, имеется возможность определять, какие действия будут появляться в раскрывающемся списке. Благодаря этой возможности вы можете обеспечить пользователей с различными ролями более гибкими страницами администрирования. Например, для модератора доски объявлений можно предусмотреть доступность таких действий, как «Unpublish content» (Снять с публикации) и «Email administrator» (Послать сообщение администратору). Администратору можно было бы дать доступ к другому представлению, включающему в список действия «Ban user» (Блокировать пользователя) и «Delete content» (Удалить материал).

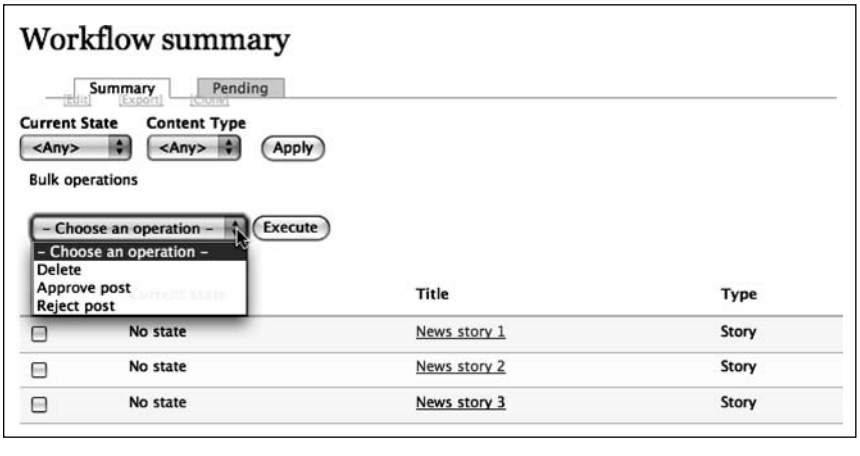


Рис. 6.15. *Представление Workflow summary (Обзор процесса), созданное с помощью модуля Views Bulk Operations*

Практика: создание страницы администрирования

Для начала, прежде чем приступить к созданию представления, нам необходимо настроить действия для одобрения и отклонения материалов с использованием наших этапов процесса публикации. Это упростит включение действий Reject (Отклонить) и Approve (Одобрить) в список модуля Views Bulk Operations.

Создание действий процесса публикации

1. Перейдите на страницу Administer (Администрирование)→Site configuration (Конструкция сайта)→Actions (Действия) (*admin/settings/actions*).

2. Выберите действие Change workflow state of post to new state... (Изменить этап процесса публикации...) в раскрывающемся списке Make a new advanced action available (Новое сложное действие) и щелкните на кнопке Create (Создать).
3. Заполните поля формы с настройками нового действия в соответствии с табл. 6.8 и сохраните ее.

Таблица 6.8. Настройки действия Approve post (Одобрить)

Параметр	Значение
Description (Описание)	Approve post (Одобрить)
Target state (Заданный этап)	Approved (Одобрено)

4. Выполните те же операции, только на этот раз для поля Description (Описание) используйте текст «Reject post» (Отклонить), а в поле Target state (Заданный этап) выберите этап Draft (Рукопись). По завершении щелкните на кнопке Save (Сохранить).

Теперь у нас имеются действия, позволяющие отклонять или одобрять материалы. Благодаря работе, проделанной ранее, всякий раз, когда будут выполняться эти действия, будут отправляться все необходимые уведомления, и в случае одобрения содержимое будет публиковаться согласно правилам триггеров и настройкам процесса публикации.

В модуле Workflow уже имеется простое представление со списком всего содержимого сайта и с информацией о том, на каком этапе в настоящий момент находится каждый из материалов. Теперь нам требуется изменить это представление, чтобы задействовать возможности модуля Views Bulk Operations.

Настройка представления

1. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Modules (Модули) (*admin/build/modules*) и включите следующие модули:
   - Other (Другой)
     – Advanced Help
   - Views
     – Views
     – Views Bulk Operations
     – Views UI
2. Если теперь вы перейдете на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Views (*admin/build/views*), то сможете увидеть в списке представление Default Node view: workflow_

summary (Представление узла по умолчанию: workflow_summary). Щелкните на ссылке Edit (Изменить) для этого представления (*admin/build/views/edit/workflow_summary*).

3. В разделе Basic settings (Базовые настройки) щелкните на параметре Style: Table (Стиль: Таблица), установите для него значение Bulk Operations (Массовые операции) и щелкните на кнопке Update (Обновить).
4. В форме Defaults: Style options (По умолчанию: Параметры стиля), которая появится ниже, в разделе Selected operations (Выбранные операции), отметьте следующие флажки и щелкните на кнопке Update (Обновить):
   - Delete (Удалить)
   - Approve post (Одобрить)
   - Reject post (Отвергнуть)
5. Щелкните на кнопке Save (Сохранить), чтобы сохранить изменения в настройках представления, которые теперь должны выглядеть, как показано на рис. 6.16.

В навигационном меню должен появиться новый пункт Workflow summary (Обзор процесса). Если перейти на страницу, изображенную на рис. 6.15, вы должны увидеть список всего содержимого на сайте с раскрывающимися списками, позволяющими фильтровать список материалов по этапу и типу содержимого. Здесь также появился новый раскрывающийся список, позволяющий выбирать операции для запуска наших действий. Отметьте любой из материалов, используя флажки, и выберите действие, которое должно быть выполнено для отмеченного узла.

Дополнительно

Итак, мы определили новый процесс публикации. Мы удовлетворили потребности наших клиентов, но далее хотим привести еще несколько модулей, достойных внимания:

Flag (http://drupal.org/project/flag)

Модуль Flag, который подробнее описывается в главе 9, можно было бы использовать, чтобы быстро одобрить статьи, не переходя на страницу редактирования или обзора.

Workflow Fields (http://drupal.org/project/workflow_fields)

Модуль Workflow Fields позволяет изменять состояние определенных полей CCK, делая их доступными только для чтения или вообще скрывая их, в зависимости от того, на каком этапе процесса публикации находится материал. Например, для сотрудников юридического отдела необходимо иметь возможность читать содержимое новой статьи и отклонять или отвергать ее, но они наверняка не должны иметь возможности вносить в нее собственные исправления.

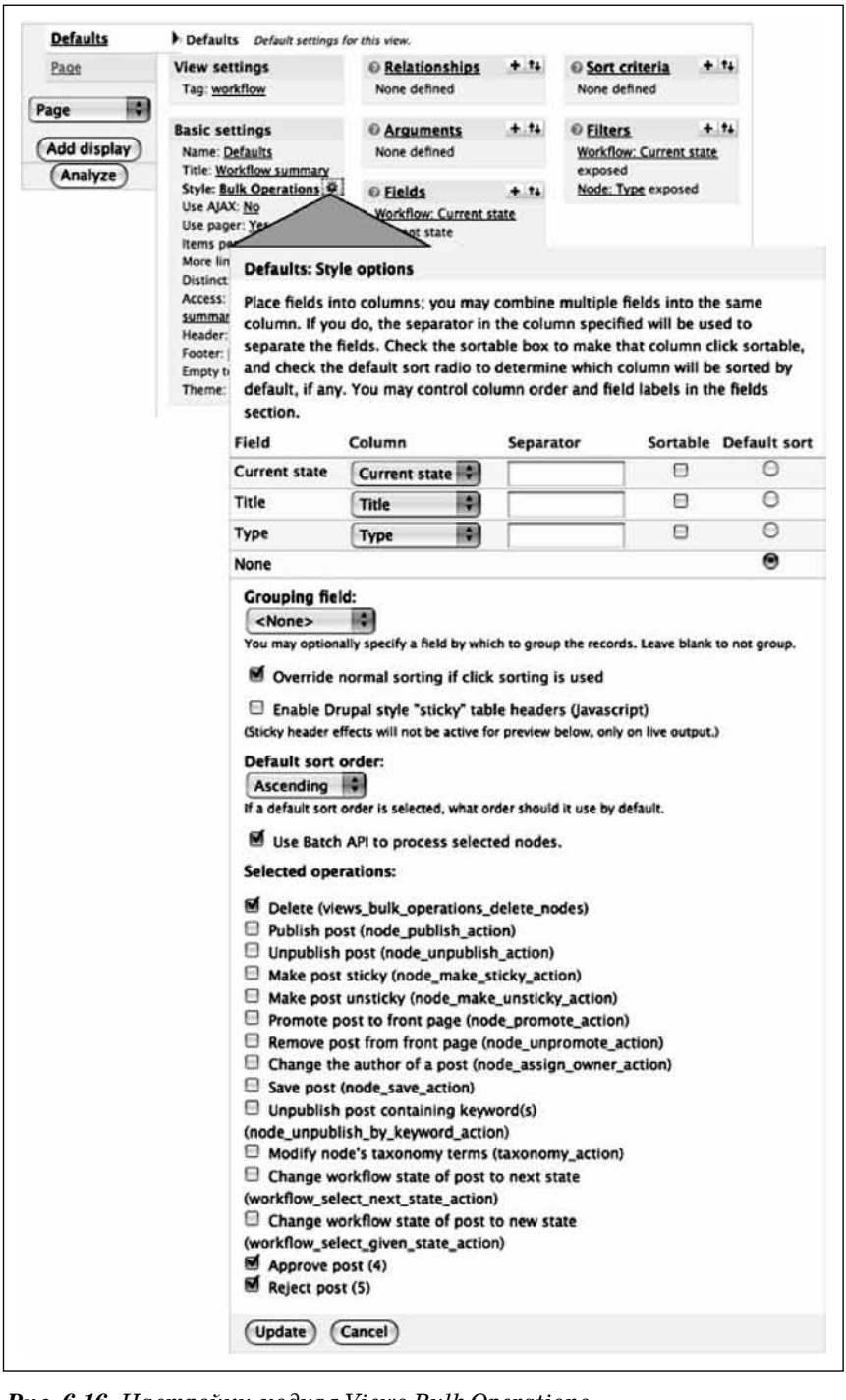


Рис. 6.16. *Настройки модуля Views Bulk Operations*

Workflow Graph (http://drupal.org/project/workflow_graph)

Этот модуль может отображать хронологию подготовки материала в виде графика, показывающего, кто из пользователей и когда переносил его на другой этап. Для простых сайтов в этом нет большой необходимости, но на крупных сайтах этот модуль может существенно упростить слежение за сложными процессами публикации.

Nodequeue (http://drupal.org/project/nodequeue)

Этот модуль часто используется на новостных сайтах, так как он позволяет отображать списки произвольных статей в порядке, указанным пользователем, например список статей, отобранных редактором. Модуль Nodequeue также интегрируется с механизмом действий, что позволяет выполнять такие операции, как автоматическое добавление новых статей в очереди по темам.

В заключение

Поздравляем! Теперь веб-сайт Twin City Arts обладает всеми основными возможностями, необходимыми сотрудникам. Мы использовали модуль Workflow для определения требуемого набора «этапов», на которых могут находиться материалы, предотвратили возможность рассмотрения редакторами статей, которые еще не были закончены авторами. Мы использовали триггеры и действия, чтобы извещать редакторов и авторов статей, когда это необходимо, и задействовали модуль Views Bulk Operations, чтобы дать редакторам возможность одновременно управлять большими группами статей. Наконец, мы использовали модуль Workspace, чтобы обеспечить авторов простым универсальным инструментом для просмотра списка их рукописей, статей, находящихся на рассмотрении, и ранее опубликованных материалов.

Ниже приводится перечень модулей, упоминавшихся в этой главе:

- Flag: *http://drupal.org/project/flag*
- Nodequeue: *http://drupal.org/project/nodequeue*
- Rules: *http://drupal.org/project/rules*
- Trigger: базовый
- Views: *http://drupal.org/project/views*
- Views Bulk Operations: *http://drupal.org/project/views_bulk_operations*
- Workflow: *http://drupal.org/project/workflow*
- Workflow Fields: *http://drupal.org/project/workflow_fields*
- Workflow Graph: *http://drupal.org/project/workflow_graph*
- Workspace: *http://drupal.org/project/workspace*

7

Фотогалерея

Как известно, лучше один раз увидеть, чем сто раз услышать, поэтому важно, чтобы сайт обладал мощными средствами для работы с изображениями. Гибкость системы Drupal обеспечивает различные способы управления изображениями и позволяет создавать как небольшие сайты, содержащие фотоальбом одного человека, так и крупные, насчитывающие миллионы пользователей, выгружающих свои фотографии.

В этой главе будут представлены следующие модули:

ImageField (http://drupal.org/project/imagefield)

Поле CCK, позволяющее выгружать изображения на сайт

ImageCache (http://drupal.org/project/imagecache)

Автоматически изменяет размеры изображений до указанных и кэширует их для последующего извлечения

Custom Pagers (http://drupal.org/project/custom_pagers)

Создает простые элементы управления постраничным выводом

Если вы собираетесь следовать за практическими упражнениями этой главы, вам следует установить Drupal, используя профиль установки Photo Gallery из примеров кода, распространяемых вместе с книгой. В результате применения этого профиля на вашем веб-сервере будет создана заготовка веб-сайта. В окончательном виде этот веб-сайт будет выглядеть, как показано на рис. 7.1 и по адресу *http://gallery.usingdrupal.com*. За дополнительной информацией об использовании профилей установки обращайтесь к разделу «Вступление».

Для выполнения практических разделов этой главы у вас должна быть настроена функция Clean URLs, и ваша версия PHP должна быть скомпилирована с поддержкой библиотеки GD. За дополнительной информацией обращайтесь к разделу «Решение проблем с модулем ImageCache» ниже в этой главе.

Рис. 7.1. *Веб-сайт фотогалереи семьи Робинсонов*

Задача

Робинсоны – это большая семья, члены которой живут далеко друг от друга и потому лишены возможности находиться в постоянном контакте. Родители долго упрашивали сына Фрица, имеющего опыт веб-разработки, создать веб-сайт, где они могли бы делиться семейными историями и выкладывать фотографии. В выходной день, когда все домашние дела были переделаны, Фриц решил создать семейный сайт фотогалереи на основе системы Drupal.

Робинсоны хотели бы позволить членам семьи присылать свои фотографии. После выгрузки на сайт фотографии должны автоматически уменьшаться в размерах для отображения на страницах, но при этом должна сохраняться возможность загружать с сайта полноразмерные версии. Им необходима возможность помещать фотографии в различные галереи и помечать их ключевыми словами. И, наконец, они хотели бы, чтобы члены семьи имели возможность комментировать фотографии.

Примечания к реализации

Вероятно, вы полагаете, что в создании сайта фотогалереи нет ничего сложного, но имея перед собой весь широкий спектр возможностей, предоставляемых Drupal, вы можете оказаться в затруднении. Но не стоит унывать, потому что если строить сайт с использованием модулей CCK и Views, такой как веб-сайт каталога вакансий в главе 3, то его

можно легко расширить до точно ориентированной на конкретные потребности галереи изображений.

Выгрузка фотографий

За работу с изображениями в системе Drupal долгое время отвечали несколько модулей, сотрудничающих (или конкурирующих) друг с другом. Универсальным решением в версиях Drupal 4.6 и 4.7 был модуль Image (*http://drupal.org/project/image*), но с появлением пакета CCK появилось несколько новых модулей, использующих этот новый, более гибкий механизм. Поскольку в своих разработках сообщество Drupal склоняется к веб-сайтам, построенным на базе модулей CCK и Views, мы также будем строить наш сайт с учетом этих новых веяний, опираясь на базовую функциональность модулей CCK и Views.

Модуль ImageField (*http://drupal.org/project/imagefield*) реализует поле CCK для выгрузки изображений, который мы можем использовать для выгрузки наших фотографий.

Создание миниатюр

Несмотря на свое название, модуль ImageCache (*http://drupal.org/project/imagecache*) обеспечивает гораздо более широкие возможности, чем простое кэширование изображений. Он часто используется в комбинации с модулем ImageField, потому что, хотя ImageField и реализует возможность выгрузки и хранения изображений, тем не менее, он не предусматривает возможности уменьшения их размеров. Модуль ImageCache способен не только создавать миниатюры, но и выполнять последовательности операций над изображениями, такие как кадрирование, вращение, масштабирование, ослабление и усиление насыщенности цвета, что обеспечивает возможность тонкой настройки отображения изображений.

Фотогалереи

Отображение изображений в виде галереи можно реализовать, создав несколько страниц с помощью модуля Views. Так как модули ImageCache и ImageField обеспечивают возможность интеграции с модулем Views, вы можете отбирать любые желаемые изображения, созданные с помощью модуля ImageField, после чего модуль ImageCache уменьшит их для отображения в представлении.

В центре внимания: модуль ImageField

Опираясь на гибкость CCK, любой модуль в системе Drupal может предоставить поле для расширения типов содержимого. Поля для текстовой или числовой информации встроены в CCK, но вы не менее просто можете добавлять поля, позволяющие выгружать аудио- и видеофайлы, переходить щелчком на ссылке и многое другое. Такие поля дополни-

тельных типов реализуются в рамках отдельных проектов на веб-сайте Drupal.org и доступны в категории «CCK» (*http://drupal.org/project/Modules/category/88*).

Модуль ImageField, доступный по адресу *http://drupal.org/project/imagefield*, реализует одно из таких специализированных полей для работы с изображениями, как показано на рис. 7.2. Как и с другими полями CCK при соответствующих настройках с полем изображения может быть связано несколько значений, что позволяет выгружать целый набор изображений в один узел. Кроме того, в тип содержимого можно добавить несколько таких полей – в случае, когда элемент содержимого должен содержать множество изображений, каждое из которых обрабатывается по-своему.

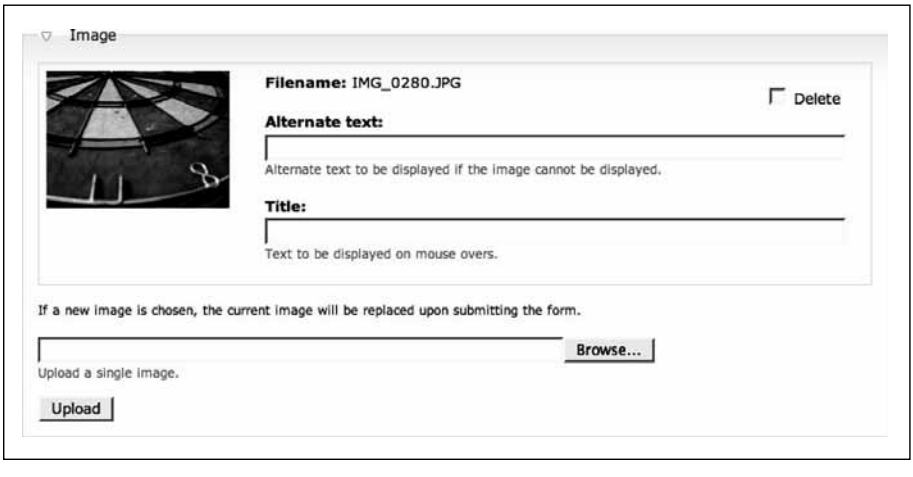


Рис. 7.2. *Поле изображения с включенными настройками «Alternate text» и «Title»*

Настройка

После включения модуль ImageField добавляет новое поле в форму настройки полей для новых или уже существующих типов содержимого. Подобно другим полям CCK поле изображения добавляется в тип содержимого на странице Administer (Администрирование)→Content management (Содержание)→Content types (Типы содержимого) (*admin/content/types*). На рис. 7.3 представлена форма настройки поля изображения.

Некоторые настройки в этой форме, такие как Label (Название), Widget type (Тип виджета), Required (Обязательно) и Number of values (Количество значений), являются общими для всех полей CCK. Эти настройки имеют для поля изображения тот же смысл, что и для других полей, и были описаны во введении в CCK в главе 3. Ниже перечислены настройки, которые являются уникальными для ImageField:

Maximum resolution (Максимальное разрешение)

Когда пользователь выгружает слишком большое изображение, это изображение будет масштабировано так, чтобы уложиться в заданное максимальное разрешение.

Minimum resolution (Минимальное разрешение)

При попытке выгрузить слишком маленькое изображение поле ImageField предложит пользователю выбрать изображение большего размера.

File path (Путь к файлам)

Этот параметр позволяет обеспечить сохранение изображений в определенном подкаталоге внутри каталога с вашими файлами. Каталог для хранения файлов можно определить на странице Administer (Администрирование)→Site Configuration (Настройка сайта)→File System (Файловая система) (*admin/settings/file-system*). Чтобы избежать складывания ваших изображений в каталог верхнего уровня, всегда следует устанавливать значение этого параметра. Часто бывает необходимо для каждого типа содержимого, определяемого на сайте, использовать отдельный каталог. Например, если имеется два типа содержимого, такие как «Article» (Статья) и «Blog post» (Сообщение в блоге), вы можете определить пути к каталогам с изображениями как *images/article* и *images/blog*, соответственно.

В случае если установлен модуль Token (*http://drupal.org/project/token*), параметр File Path (Путь к файлам) можно сделать динамическим. Смотрите описание примера использования модуля Token в паре с модулем Pathauto в главе 5.

Title (Заголовок) / ALT text (Альтернативный текст)

Если настройка включена, то она сделает доступными пользователю дополнительные поля для ввода значений атрибутов *alt* и *title* изображения. Значения, введенные пользователем, выделены жирным шрифтом в следующем примере кода разметки:

```
<img src="/files/images/sample.png" alt="Alternate text displayed here"
title="Title text displayed here" width="200" height="100" />
```

Обычно текст в атрибуте *alt* никогда не отображается перед конечным пользователем, но он имеет большое значение для обеспечения доступности, так как его используют устройства чтения с экрана и другое вспомогательное программное обеспечение. Содержимое атрибута *title* должно отображаться в виде всплывающей подсказки при наведении указателя мыши на изображение.

Default list value (Перечисляется по умолчанию) / How should the list value be handled? (Как обрабатывается перечисление по умолчанию?)

Когда для отображения изображения используется формат «Generic file display» (Универсальный формат отображения файлов), изобра-

Photo settings

These settings apply only to the *Image* field as it appears in the *Photo* content type.

Help text:

Upload a single image.

Instructions to present to the user below this field on the editing form.
Allowed HTML tags: <a> <b> <big> <code> <del> <i> <ins> <pre> <q> <small>   <sub> <sup> <tt>   <p>

Permitted upload file extensions.:

jpg jpeg png gif

Extensions a user can upload to this field. Seperate extensions with a space and do not include the leading dot.

Maximum resolution for Images:

0

The maximum allowed image size expressed as WIDTHxHEIGHT (e.g. 640x480). Set to 0 for no restriction. If a larger image is uploaded, it will be resized to reflect the given width and height.

Minimum resolution for Images:

0

The minimum allowed image size expressed as WIDTHxHEIGHT (e.g. 640x480). Set to 0 for no restriction. If an image that is smaller than these dimensions is uploaded it will be rejected.

File size restrictions

Limits for the size of files that a user can upload. Note that these settings only apply to newly uploaded files, whereas existing files are not affected.

Maximum upload size per file:

Specify the size limit that applies to each file separately. Enter a value like "512" (bytes), "80K" (kilobytes) or "50M" (megabytes) in order to restrict the allowed file size. If you leave this this empty the file sizes will be limited only by PHP's maximum post and file upload sizes.

Maximum upload size per node:

Specify the total size limit for all files in field on a given node. Enter a value like "512" (bytes), "80K" (kilobytes) or "50M" (megabytes) in order to restrict the total size of a node. Leave this empty if there should be no size restriction.

Path settings

Title text settings

ALT text settings

Global settings

These settings apply to the *Image* field in every content type in which it appears.

☑ Required

Number of values:

1

Select a specific number of values for this field, or 'Unlimited' to provide an 'Add more' button so the users can add as many values as they like.
Warning! Changing this setting after data has been created could result in the loss of data!

Default list value: *

◉ Listed

○ Hidden

The list option determines whether files are visible on node views. This will be used as the default value for the list option.

How should the list value be handled?: *

○ User Configurable. (Users will be able to set the list value per file.)

◉ Enforce Default. (The default list value will be used for all files, and the list checkbox will not be displayed to users.)

Рис. 7.3. *Форма с настройками поля изображения*

жения отображаются в виде списка. Если параметр Default list value (Перечисляется по умолчанию) имеет значение Listed (Перечисляется), выгружаемые изображения будут включаться в список. Этот параметр не учитывается, если для отображения изображений используется формат, отличный от «Generic file display» (Универсальный формат отображения файлов).

Если параметр Default list value (Перечисляется по умолчанию) имеет значение User configurable (Определяется пользователем), пользователям будет предоставлена возможность определять значение этого параметра при выгрузке файла. С настройками по умолчанию список изображений отображается, как показано на рис. 7.4.

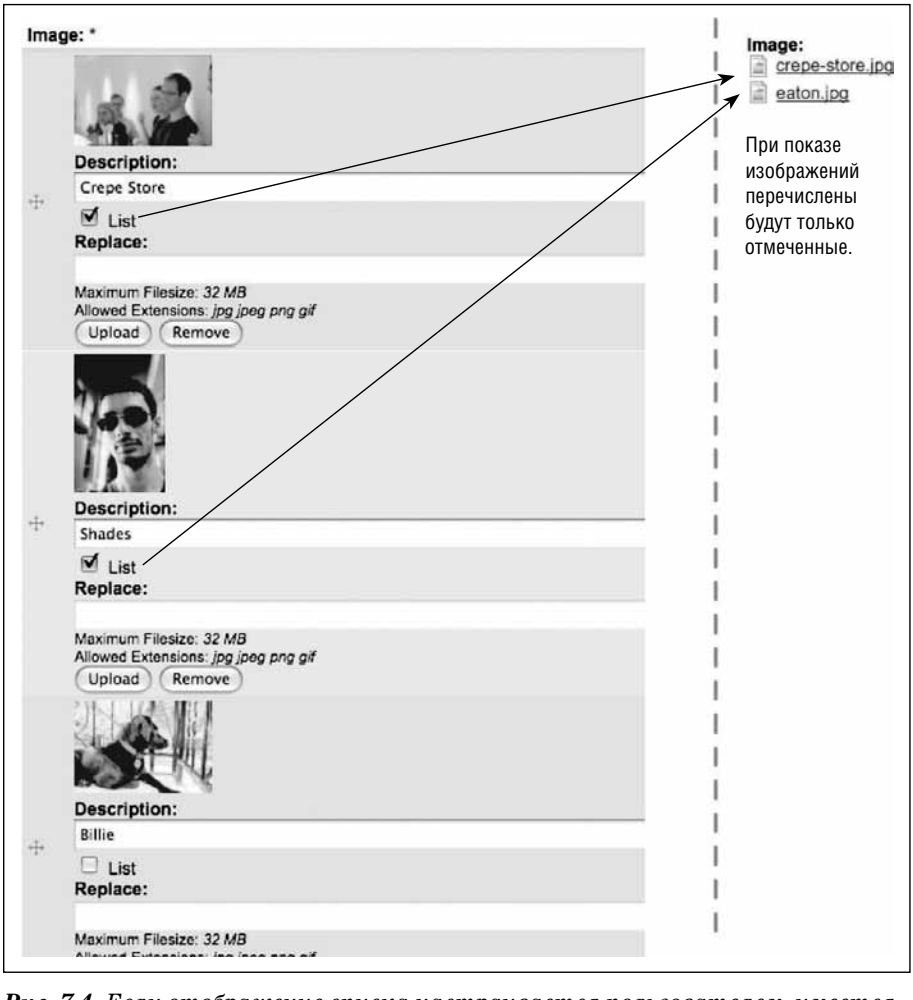


Рис. 7.4. *Если отображение списка настраивается пользователем, имеется возможность указать, должно ли изображение присутствовать в списке*

Практика: выгрузка фотографий

Для начала нам необходимо заняться основами: создать форму для загрузки изображений и определить некоторые рубрики таксономии, которые будут использоваться для навигации по фотогалерее.

Тип содержимого Photo

Первое, что нам необходимо сделать, – это создать новый тип содержимого для фотографий с именем Photo (Фотография). Этот новый тип даст пользователям возможность выгружать полноразмерные фотографии с помощью модуля ImageField.

1. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Modules (Модули) (*admin/build/modules*) и включите следующие модули:
   - CCK
     - Content
     - FileField
     - ImageField
   - Core – optional (Ядро – дополнительные)
     - Taxonomy
2. Перейдите на страницу Administer (Администрирование)→Content management (Содержание)→Content types (Типы содержимого) (*admin/content/types*) и добавьте новый тип содержимого, щелкнув на вкладке Add content type (Добавить тип содержимого). Создайте тип содержимого с названием *Photo* (Фотография), используя настройки из табл. 7.1.

Таблица 7.1. Настройки для нового типа содержимого Photo (Фотография)

Параметр	Значение
Identification (Идентификация)	
Name (Имя)	Photo (Фотография)
Type (Тип)	photo
Description (Описание)	A photo containing a Robinson family member or friend (Фотография члена семьи Робинсон или их друга)
Submission form settings (Свойства формы)	
Title field label (Название поля заголовка)	Title (Заголовок)
Body field label (Название поля текста)	Caption (Подпись)

3. После создания типа Photo (Фотография) вы вернетесь обратно, на страницу со списком типов содержимого. Теперь нужно добавить поле изображения в новый тип Photo (Фотография). Щелкните на ссылке manage fields (управлять полями) для типа Photo (Фотография) (*admin/content/node-type/photo/add_field*) и введите значения параметров в соответствии с табл. 7.2.

Таблица 7.2. Настройки, определяемые при добавлении поля photo в тип содержимого Photo (Фотография)

Параметр	**Значение**
Label (Название)	Photo (Фотография)
Field name	photo
Select a field type (Выберите тип поля)	Image (Изображение)
Select a widget type (Выберите тип виджета)	Image (Изображение)

4. После создания нового поля на экране появится форма настройки (*admin/content/node-type/photo/fields/field_photo*). Введите значения параметров в соответствии с табл. 7.3. Имена большинства параметров говорят сами за себя, однако параметр, определяющий максимальное разрешение изображений, имеет специальное значение 0, означающее отсутствие ограничений на величину разрешения выгружаемого изображения. Так как одно из пожеланий семьи состоит в том, чтобы обеспечить возможность обмена полноразмерными фотографиями, мы оставим в этом параметре значение по умолчанию 0. Все изображения мы будем выгружать в подкаталог с именем *photos*.

Таблица 7.3. Настройки поля photo в типе содержимого Photo (Фотография)

Параметр	**Значение**
Photo settings (Настройки изображения)	
Permitted upload file extensions (Допустимые расширения выгружаемых файлов)	jpg jpeg png gif
Maximum resolution for images (Максимальное разрешение для изображений)	0
Minimum resolution for images (Минимальное разрешение для изображений)	0

Path settings > File path (Настройки пути > Путь к файлу)	photos
Global settings (Общие настройки)	
Required (Обязательно)	отмечено
Number of values (Количество значений)	1
Default list value (Перечисляется по умолчанию)	Listed (Перечисляется)
How should the list value be handled? (Как обрабатывается перечисление по умолчанию?)	Enforce Default (Принудительно использовать значение по умолчанию)
Description Field (Поле описания)	Disabled (Отключено)

В случае если модуль Token включен, путь к каталогу, где должны сохраняться выгружаемые изображения, можно определять динамически, например на основе имени пользователя или даты.

5. После сохранения настроек вы вернетесь обратно, на страницу со списком всех полей типа содержимого Photo (Фотография). Расположите поля в следующем порядке и щелкните на кнопке Save (Сохранить):
   - Title (Заголовок)
   - Photo (Фотография)
   - Caption (Подпись)
   - Menu settings (Настройки меню)
6. На этом настройку типов содержимого можно считать законченной, но нам необходимо разрешить пользователям выгружать изображения на сайт. Перейдите на страницу Administer (Администрирование)→User management (Управление пользователями)→Permissions (Разрешения) (*admin/user/permissions*), где вы сможете определить разрешения для каждой имеющейся роли. Определите разрешения в соответствии с табл. 7.4.

Закончив настройку типа содержимого Photo (Фотография), попробуйте выгрузить новую фотографию, перейдя на страницу Create content (Создать материал)→Photo (Фотография) (*node/add/photo*). Обычный пользователь должен увидеть форму, как показано на рис. 7.5.

Таблица 7.4. Разрешения на доступ к новому типу содержимого Photo (Фотография)

Разрешение	anonymous user (анонимный пользователь)	authenticated user (зарегистрированный пользователь)	editor (редактор)	site administrator (администратор сайта)
модуль node				
create photo content (создавать материал типа photo)		отмечен		
delete any photo content (удалять любой материал типа photo)			отмечен	отмечен
delete own photo content (удалять собственный материал типа photo)		отмечен		
edit any photo content (изменять любой материал типа photo)			отмечен	отмечен
edit own photo content (изменять собственный материал типа photo)		отмечен		

Галереи изображений

Несмотря на то, что возможность выгружать фотографии уже позволяет семье Робинсон приступить к использованию сайта, тем не менее, Фриц понимает, что на любом сайте, где пользователи делятся друг с другом фотографиями, необходимо организовать галереи, чтобы пользователи могли подразделять фотографии по категориям. В случае нашей фотогалереи Фриц определил, что галерея – это не более, чем группа фотографий, объединенных общим названием или относящихся к одной категории. Такая организация в точности может быть реализована с помощью встроенного модуля Taxonomy, представленного в главе 2. Мы можем создать несколько предопределенных галерей, определив словарь с названием каждой из них.

Кроме того, многие сайты с фотогалереями дают пользователям возможность относить свои фотографии к любой произвольной категории, ука-

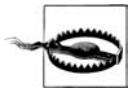
Рис. 7.5. *Форма отправки фотографии, как ее видит пользователь, наделенный ролью authenticated user (зарегистрированный пользователь)*

зывая произвольное ключевое слово (это называется произвольной маркировкой (free tagging)). Подобная возможность классификации также поддерживается встроенным модулем Taxonomy.

1. Перейдите на страницу Administer (Администрирование)→Content management (Содержание)→Taxonomy (Таксономия) (*admin/content/taxonomy*).
2. Щелкните на вкладке Add vocabulary (Добавить словарь) и введите значения параметров из табл. 7.5 в форму настройки словаря.

Таблица 7.5. Значения параметров настройки нового словаря Gallery (Галерея)

Параметр	Значение
Identification (Идентификация)	
Vocabulary name (Название словаря)	Gallery (Галерея)
Content types (Типы материалов)	
Photo (Фотография)	отмечено
Submission form settings (Свойства формы)	
Required (Обязательно)	отмечено
Weight (Вес)	–1



Значение веса словаря Gallery (Галерея) должно быть меньше веса словаря Tags (Теги), чтобы термины из словаря Gallery (Галерея) предшествовали терминам из словаря Tags (Теги). Это будет важно, когда позднее мы будем настраивать собственный компонент постраничного просмотра Custom Pager.

3. Вернувшись на страницу настройки модуля Taxonomy, щелкните на вкладке Add vocabulary (Добавить словарь) еще раз, чтобы создать словарь произвольной маркировки, и установите значения параметров в соответствии с табл. 7.6.

Таблица 7-6. Значения параметров настройки нового словаря Tags (Теги)

Параметр	Значение
Identification (Идентификация)	
Vocabulary name (Название словаря)	Tags (Теги)
Content types (Типы материалов)	
Photo (Фотография)	отмечено
Submission form settings (Свойства формы)	
Tags (Теги)	отмечено
Weight (Вес)	0

4. Новый словарь Tags (Теги) позволит добавлять новые рубрики непосредственно в форме создания узла, в то время как словарь Gallery (Галерея) будет отображать названия галерей, определенные администратором сайта. Щелкните на ссылке add terms (добавить термины) для словаря Gallery (Галерея), чтобы добавить несколько названий первых галерей. Единственный параметр, значение которого необходимо определить, – это параметр Term (Термин). Создайте несколько терминов, аналогичных предложенным в табл. 7.7 (вы можете использовать любые термины, по своему желанию).

Таблица 7.7. Примеры терминов (названий галерей) для словаря Gallery (Галерея)

Название термина
Sunnyvale family picnic (Пикник семьи Саннивейл)
Barcelona vacation (Каникулы в Барселоне)
Boston road trip (Поездка в Бостон)
Charlie's distraction (Развлечения Чарли)

Теперь Робинсоны смогут выбирать название галереи при выгрузке или редактировании фотографий. Форма изменения узла с фотографиями на странице Create Content (Создать материал)→Photo (Фотография) (*node/add/photo*), теперь должна содержать раскрывающийся список с названиями галерей и с текстовым полем для ввода тегов, как показано на рис. 7.6.

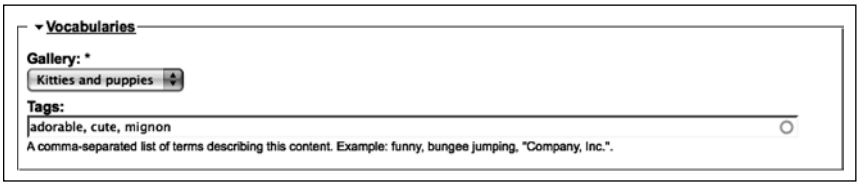


Рис. 7.6. *Поля для выбора названия галереи и добавления тегов*

В центре внимания: модуль ImageCache

Если на предыдущем практическом этапе вы выгружали достаточно большие изображения, весьма вероятно, что вы нарушили схему расположения элементов на странице сайта. Чтобы этого избежать, следует изменить масштаб этих изображений с целью привести их к подходящему размеру, а для страниц со списками создать миниатюры. Модуль ImageCache предоставляет возможность решить эту и другие задачи, связанные с отображением изображений.

Что касается работы с файлами, то модуль ImageCache – один из самых богатых возможностями модулей в системе Drupal. Несмотря на свое название, модуль ImageCache может выполнять множество других операций помимо кэширования изображений. Фактически он является полноценным инструментом для работы с изображениями, который, скорее всего, станет основой механизма обработки изображений в будущих версиях ядра Drupal.

В число операций над изображениями, которые обеспечиваются модулем ImageCache, входят такие действия, как кадрирование, масштабирование или изменение размеров, допускающие возможность объединения в комбинации, которые называются *наборами правил* (*image preset*). Комбинируя операции, можно создавать собственные варианты отображения изображений. На рис. 7.7 показаны результаты применения набора правил, объединяющего в себе операции кадрирования и вращения, с целью получить изображение квадратной формы, повернутое на 90 градусов.

После выполнения операций над изображением в игру вступает часть «cache» (кэш) из названия модуля ImageCache. Изображение сначала создается, а затем сохраняется в каталоге с названием, совпадающим с названием набора правил для ImageCache. Благодаря этому операции над изображениями выполняются всего один раз, а результаты сохра-

няются на диске. На рис. 7.8 изображена схема процесса работы модуля ImageCache, выполняющего запрос на получение изображения.

Рис. 7.7. *Набор правил, в котором используются две операции: масштабирование с кадрированием и вращение*

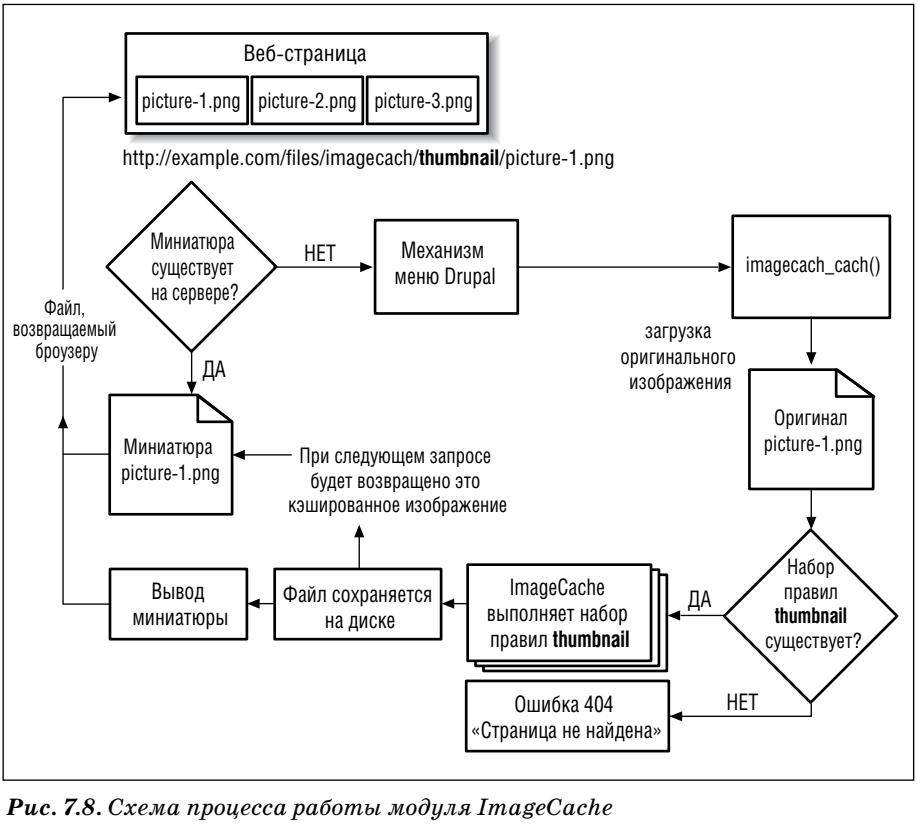


Рис. 7.8. *Схема процесса работы модуля ImageCache при создании миниатюры*

Наборы правил и операции

На рис. 7.9 изображена главная панель управления модулем ImageCache на странице Administer (Администрирование)→Site building (Конструкция сайта)→ImageCache (*admin/build/imagecache*), где присутствует список наборов правил, доступных на сайте. Названия наборов правил не имеют иного значения, кроме как описание операций, выполняемых над изображениями. Вы можете добавлять, изменять или удалять наборы правил, а также «очищать» их. Ссылка Flush (Очистить) позволяет удалить все изображения, созданные с помощью набора правил. Так как изображения генерируются «на лету», вы в любой момент можете удалить все миниатюры на сайте – они будут созданы при первой же попытке получить их. Вы можете потребовать очистить кэш изображений, созданных наборами правил, если каким-то образом миниатюры окажутся старше оригиналов.

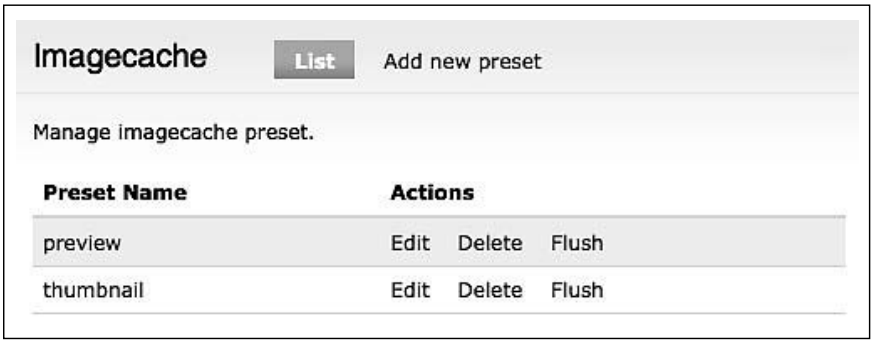


Рис. 7.9. *Страница управления наборами правил модуля ImageCache*

Название набора правил становится частью URL для всех создаваемых изображений, поэтому желательно давать короткие названия, содержащие только символы нижнего регистра и использовать в названиях только алфавитно-цифровые символы, подчеркивания и дефисы. Если на сайте предполагается использовать стандартные размеры для изображений, будет неплохо, если названия наборов правил будут включать и размеры, например «160_square», «200_width» или «300×200_resize». В наших примерах мы используем такие названия, как «thumbnail» и «preview», описывающие семантику их использования.

Но самый интересный момент наступает, когда возникает необходимость добавить в набор правил новые операции. Один набор может состоять из множества операций, при этом операции будут выполняться в порядке сверху вниз. Всякий раз, когда производится изменение набора правил, кэшированные файлы удаляются, чтобы они могли быть созданы заново. Это упрощает, например, переход от миниатюр с размером 100 пикселов к миниатюрам с размером 120 пикселов (или ввод

в действие любых других изменений). На рис. 7.10 показана форма добавления новых операций в набор.

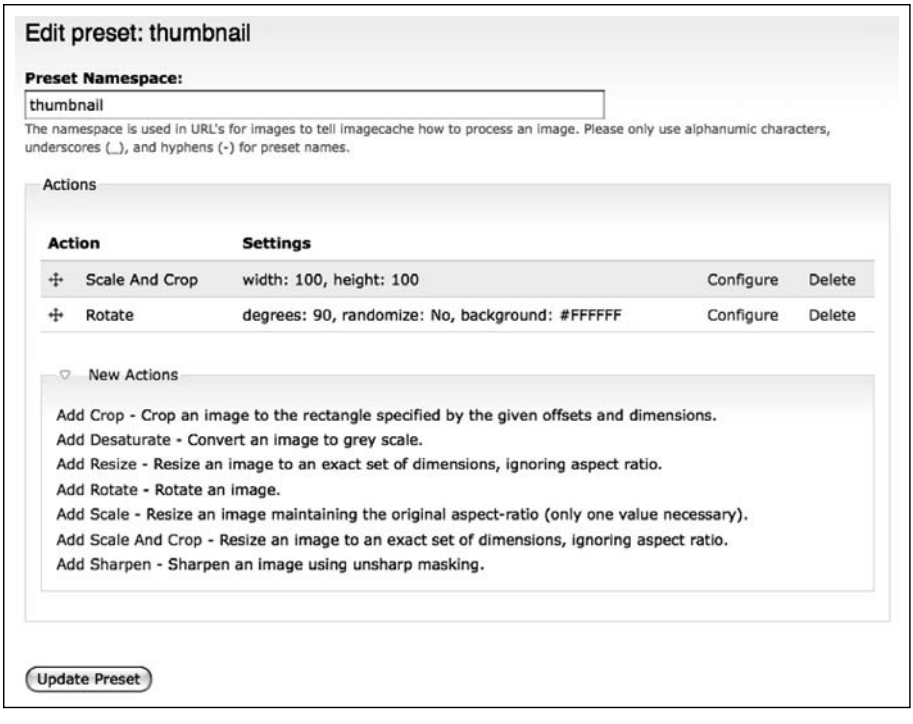


Рис. 7.10. *Операции в наборе правил для модуля ImageCache*

Помимо веса каждая операция имеет свои уникальные настройки. Мы опишем только самые основные операции: crop (кадрирование), resize (изменение размеров), scale (масштабирование) и scale and crop (масштабирование с кадрированием).

Кадрирование

Операция crop позволяет обрезать края изображения. Онаможет принимать значения в пикселах, но чаще в качестве ее значений используются ключевые слова *top*, *right*, *bottom*, *left* и *center*. Конечный результат операции кадрирования будет напоминать изображение на рис. 7.11.

Изменение размеров

Операция resize может использоваться для изменения размеров изображения до заданных значений. Новая ширина и высота могут задаваться целыми значениями, чтобы выполнить масштабирование до указанных размеров в пикселах, или включать знак процента – для выполнения масштабирования в процентном отношении.

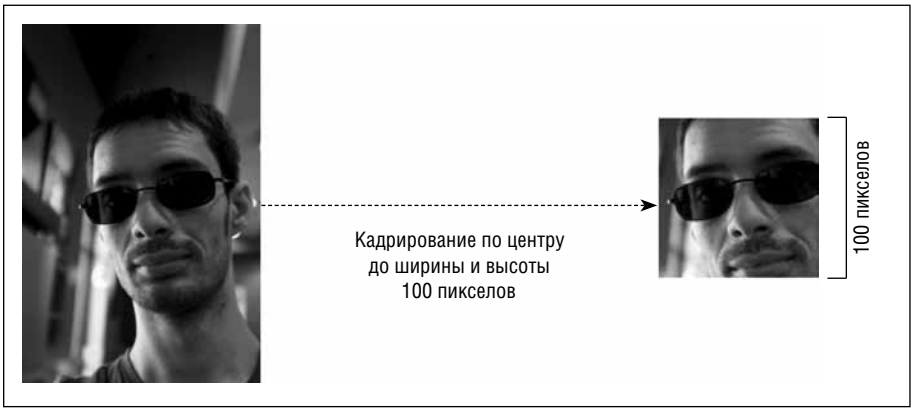


Рис. 7.11. *Кадрирование с обрезанием краев изображения*

Обычно вместо операции resize используется операция scale, так как простая операция изменения размеров может приводить к изменению пропорций изображения. Операция resize не заботится о сохранении пропорций и изменяет размеры изображения точно до указанных размеров, как показано на рис. 7.12.

Рис. 7.12. *Применение операции resize может приводить к нарушению пропорций изображения*

Масштабирование

Операция scale используется для пропорционального изменения размеров изображения. В отличие от операции resize, для операции scale достаточно указать либо ширину, либо высоту. Размер, для которого не указано значение, будет определяться исходя из масштаба изменения заданного размера. Если заданы оба размера, масштабирование изображения будет выполняться так, чтобы оба его размера оказались меньше или равны указанным значениям.

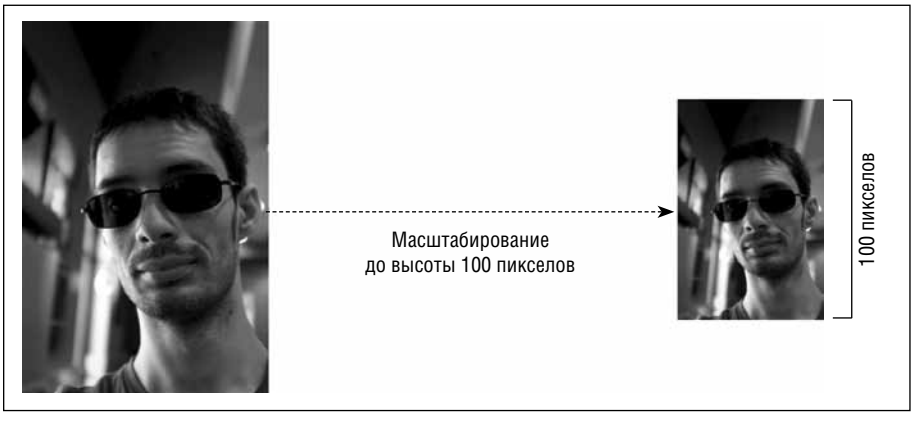


Рис. 7.13. *Масштабирование изображения с сохранением пропорций*

Если вам требуется, чтобы изображения на сайте были не меньше определенного размера, можно задействовать параметр Allow Upscaling (Разрешить масштабирование вверх), чтобы изображения могли иметь размеры больше указанных.

Операция масштабирования всегда будет стремиться сохранить пропорции оригинального изображения. Окончательный результат операции масштабирования показан на рис. 7.13.

Масштабирование с кадрированием

Как следует из названия, операция scale and crop – это комбинация операций масштабирования и кадрирования. Эта операция сначала выполняет масштабирование изображения до указанного размера, а затем выполняется кадрирование (эта операция также называется *zoom crop*). Эта операция часто бывает удобна для создания квадрат-

Рис. 7.14. *Операция scale and crop выполняет кадрирование по большей стороне, сохраняя пропорции*

ных миниатюр, сохраняющих при этом пропорции оригинальных изображений. Пример действия операции масштабирования с кадрированием показан на рис. 7.14.

В модуле ImageCache доступно больше операций, чем было перечислено, и постоянно добавляются новые. В число других операций, которые вы можете использовать, входят: поворот, водяные знаки, рамка, добавление текстовых надписей, изменение яркости и прозрачности. Полный список операций можно найти на странице проекта ImageCache по адресу *http://drupal.org/project/imagecache*. Расширенный набор операций можно получить, установив модуль ImageCache Actions, доступный по адресу *http://drupal.org/project/imagecache_actions*.

Использование наборов правил

После создания наборов правил на странице администрирования модуля ImageCache необходимо сообщить системе Drupal, где должны использоваться эти наборы. Модули ImageField и FileField обеспечивают возможность отображения изображений в полный размер или в виде списка файлов. После создания наборов правил ImageCache появляются дополнительные способы отображения изображений.

Форматы CCK

Как правило, настройка отображения изображений производится с помощью форматов CCK, как показано на рис. 7.15. Для каждого набора правил модуль ImageCache добавляет четыре новых формата:

[название набора] image linked to node (изображение со ссылкой на узел)

Отображает изображение с размерами, определяемыми набором правил, ссылающееся на «родительский» узел.

[название набора] image linked to image (изображение со ссылкой на изображение)

Отображает изображение с размерами, определяемыми набором правил, ссылающееся на на оригинальную версию изображения.

[название набора] file path (путь к файлу)

Отображает только путь к изображению, обработанному данным набором правил. Используется для отладки.

[название набора] URL

Отображает только адрес URL изображения, обработанного данным набором правил. Используется для отладки.

Форматы CCK, определяемые модулем ImageCache, также доступны модулю Views, что позволяет создавать такие отображения, как таблица миниатюр. Более подробно об этом будет рассказываться в разделе «Практика: модуль ImageCache» ниже в этой главе.

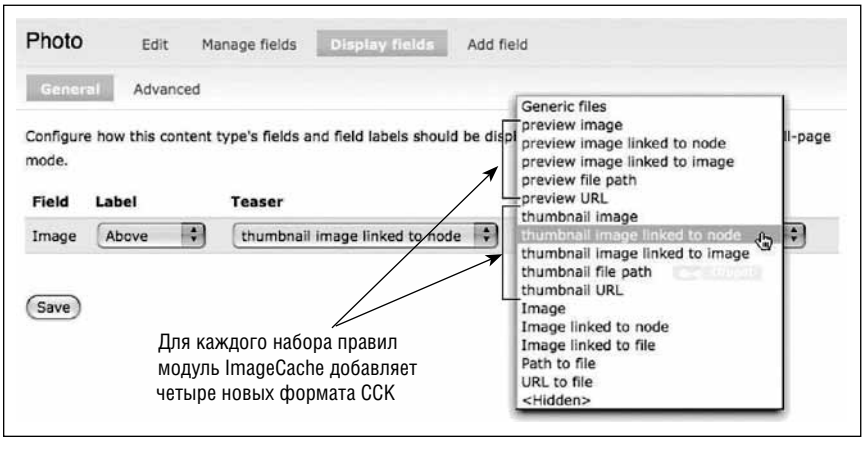


Рис. 7.15. *Настройка поля изображения с использованием набора правил ImageCache*

Просмотр вручную изображения, обработанного набором правил

Существует возможность в любой момент просмотреть оригинальное изображение и изображение, обработанное набором правил, вручную собрав адрес URL. Порядок сборки URL изображен на рис. 7.16.

После создания набора правил для модуля ImageCache легко можно проверить, как будет выглядеть изображение, вручную набрав адрес URL изображения.

Полное описание функций тем и порядок их использования приводится в главе 11.

Реализация отображения изображений, созданных средствами ImageCache, на языке PHP

Когда возникает необходимость реализовать отображение изображений, созданных средствами модуля ImageCache, внутри темы или модуля, самый простой и лучший способ заключается в использовании следующих функций:

- *imagecache_create_url()*: Возвращает полный адрес URL изображения

```
print imagecache_create_url('my_preset', $path)
```

- *theme_imagecache()*: Выводит разметку HTML для отображения изображения

```
print theme('imagecache', 'my_preset', $path, $alt, $title,
$attributes);
```

Оригинальный файл

http://localhost/sites/default/files/images/picture-1.jpg

Каталог для файлов | Путь к изображению

Файл, созданный модулем ImageCache

http://localhost/sites/default/files/imagecash/**thumbnail**/images/picture-1.jpg

Каталог для файлов | Название набора правил | Путь к изображению

Рис. 7.16. *Сборка адреса URL для заданного набора правил ImageCache*

Разрешение проблем с модулем ImageCache

Модуль ImageCache использует некоторые расширенные особенности языка PHP и веб-сервера Apache, такие как перезапись адресов URL и библиотека GD для работы с изображениями. Так как требования программного обеспечения модуля ImageCache очень строги, то любые ошибки в настройке веб-сервера или Drupal могут приводить к нарушениям в его работе. В следующих разделах описываются типичные проблемы, которые приходится решать, чтобы добиться нормальной работы модуля ImageCache.

Проверка функции Clean URLs

Чаще всего проблемы возникают из-за отключенной функции Clean URL (или она не поддерживается программным обеспечением веб-сервера). Зайдите на страницу Administer (Администрирование)→Site configuration (Настройка сайта)→Clean URLs (Чистые ссылки) (*admin/settings/clean-urls*), как показано на рис. 7.17. Если перед вами появится сообщение об ошибке, обращайтесь к странице руководства с описанием настройки функции Clean URLs на веб-сервере (*http://drupal.org/node/15365*).

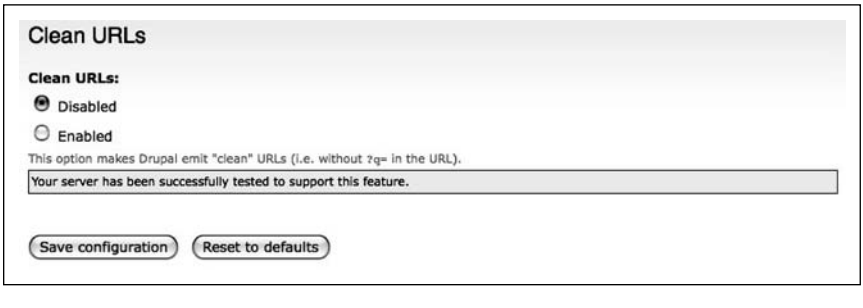
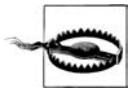


Рис. 7.17. *Функция Clean URLs не может быть включена, пока проверка не увенчается успехом*



Функция Clean URLs в Drupal требует наличия расширения mod_rewrite веб-сервера Apache. Если система Drupal работает под управлением веб-сервера Microsoft IIS, то для корректной работы модуля ImageCache необходимо установить стороннее решение, выполняющее перезапись адресов URL.

Проверка наличия библиотеки GD

Еще одна типичная проблема – отсутствие на сервере библиотеки GD для работы с изображениями. Она может быть причиной того, что при переходе вручную по адресу URL, соответствующему набору правил для модуля ImageCache, изображение отсутствует. Библиотека GD – это пакет программного обеспечения, который по умолчанию устанавливается вместе с PHP, но иногда, при нестандартной установке PHP, она может исключаться из установки. Проверить наличие библиотеки GD можно, перейдя на страницу Administer (Администрирование)→Reports (Отчеты)→Status report (Отчет о состоянии) (*admin/reports/status*). Здесь вы должны увидеть сообщение, как показано на рис. 7.18, подтверждающее наличие библиотеки GD.

Рис. 7.18. *Сообщение о наличии библиотеки GD на странице Status report (Отчет о состоянии)*

Если библиотека GD присутствует, но вы по-прежнему не видите изображений, сгенерированных модулем ImageCache, попробуйте проверить всю конфигурацию PHP, создав и запустив на сервере файл *info.php*, содержащий всего одну строчку кода:

```
<?php phpinfo(); ?>
```

Кроме того, краткий список сведений о PHP можно получить непосредственно в административном разделе Drupal, перейдя на страницу Administer (Администрирование)→Reports (Отчеты)→Status report и щелкнув на ссылке с номером версии PHP (*admin/reports/status/php*). Несмотря на то, что этот список не так полон, как информация, возвращаемая файлом *info.php*, она может быть полезна для получения общих сведений о настройках PHP на сервере.

Проверьте раздел с настройками библиотеки GD, который должен выглядеть, как показано на рис. 7.19. Убедитесь в наличии поддержки всех типов файлов изображений, которые будут выгружаться на сервер.

По окончании работы с файлом *info.php* удалите его. Он предоставляет слишком много информации о настройках сервера, что может подтолкнуть на неблаговидные поступки тех, чьи помыслы далеки от благородных.

Если этот раздел полностью отсутствует на странице, это свидетельствует о том, что библиотека GD вообще не установлена.

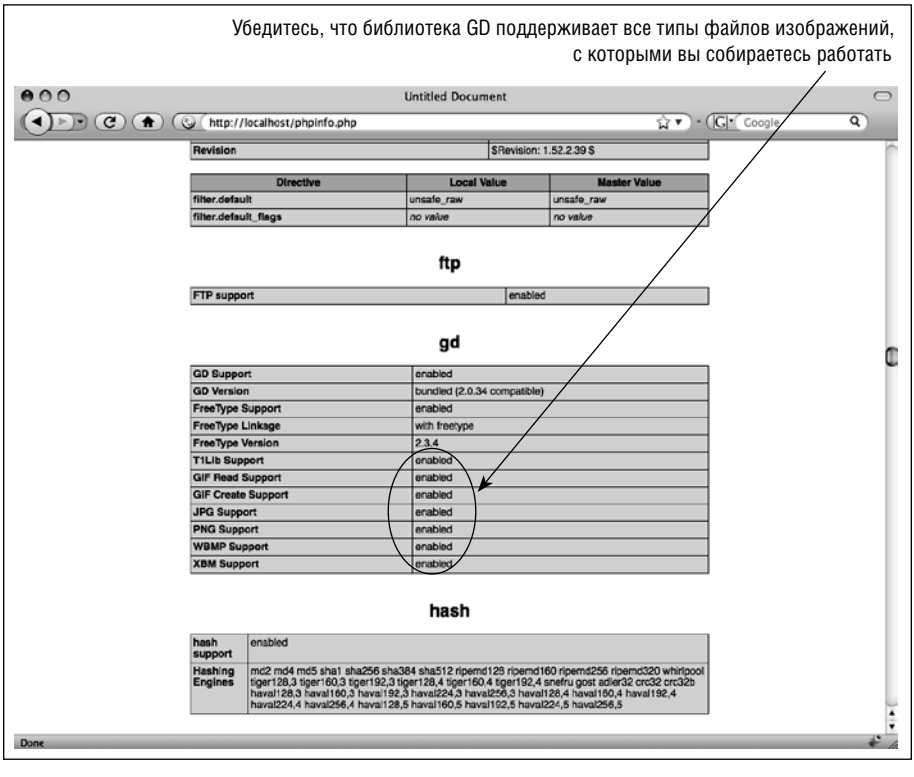


Рис. 7.19. *Проверка наличия библиотеки GD в установке PHP*

Практика: модуль ImageCache

К настоящему моменту мы имеем возможность выгружать и классифицировать фотографии, однако в списках отображаются полноразмерные фотографии, что не совсем то, что требуется Робинсонам! Нам необходимо создать уменьшенные копии изображений, оставляя при этом оригинальные изображения нетронутыми. Для начала нам необходимо включить модуль ImageCache, чтобы обеспечить возможность получения миниатюр, а затем настроить наш тип содержимого Photo (Фотография) так, чтобы он использовал эти миниатюры:

1. Проверьте наличие поддержки функции Clean URLs на странице Administer (Администрирование)→Site configuration (Настройка сайта→Clean URLs (Чистые ссылки) (*admin/settings/clean-urls*).
2. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Modules (Модули) (*admin/build/modules*) и включите следующие модули:

- ImageCache
  - ImageAPI
  - ImageAPI GD2
  - ImageCache
  - ImageCache UI

Создание наборов правил для ImageCache

Определим пару наборов правил для создания миниатюр: первый набор будет генерировать «миниатюры», имеющие форму квадрата со стороной 120 пикселов, а второй – «изображения для предварительного просмотра», имеющие больший размер и предназначенные для отображения на страницах с фотографиями (эти изображения будут иметь ширину не более 480 пикселов, чтобы не нарушить взаимное расположение элементов страниц):

1. Перейдите на страницу настройки модуля ImageCache Administer (Администрирование)→Site building (Конструкция сайта)→ImageCache (*admin/build/imagecache*). Щелкните на вкладке Add new preset (Добавить новый набор правил).
2. Введите название набора thumbnail (миниатюра) в поле Preset namespace (Название набора правил) и щелкните на кнопке Create New Preset (Создать новый набор правил). После этого откроется страница настройки набора правил, как показано на рис. 7.20.

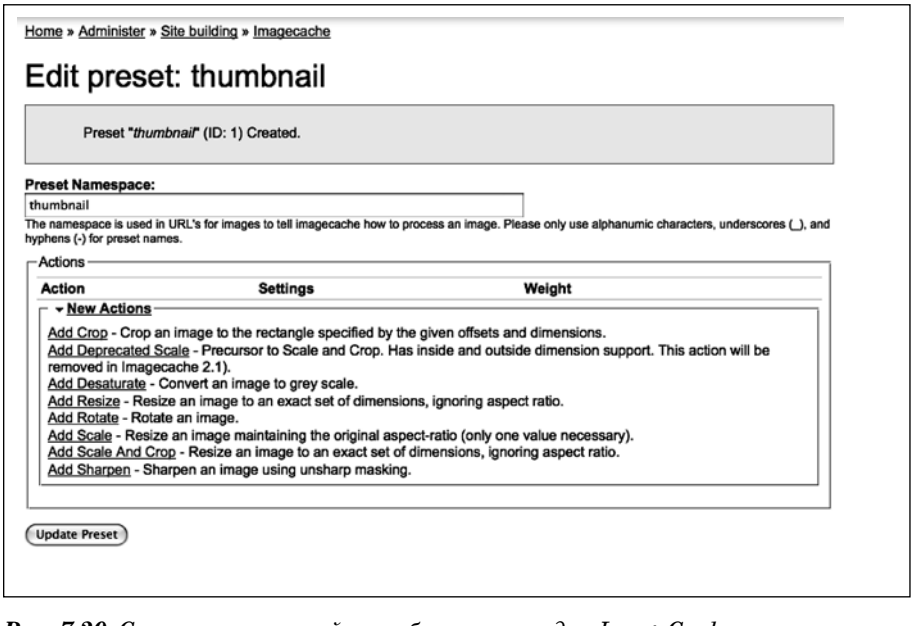


Рис. 7.20. *Страница настройки набора правил для ImageCache*

3. Ниже на странице выполняется настройка набора правил. В группе параметров New Actions (Новые действия) щелкните на ссылке Add Scale and Crop (Добавить масштабирование с кадрированием). Введите настройки операции масштабирования с кадрированием в соответствии с табл. 7.8, как показано на рис. 7.21, чтобы определить набор правил, создающий миниатюры, имеющие форму квадрата со стороной 120 пикселов. По окончании щелкните на кнопке Add Action (Добавить действие).

Таблица 7.8. Настройка операции масштабирования с кадрированием для набора правил «thumbnail» (миниатюра)

Параметр	Значение
Weight (Вес)	0
Width (Ширина)	120
Height (Высота)	120

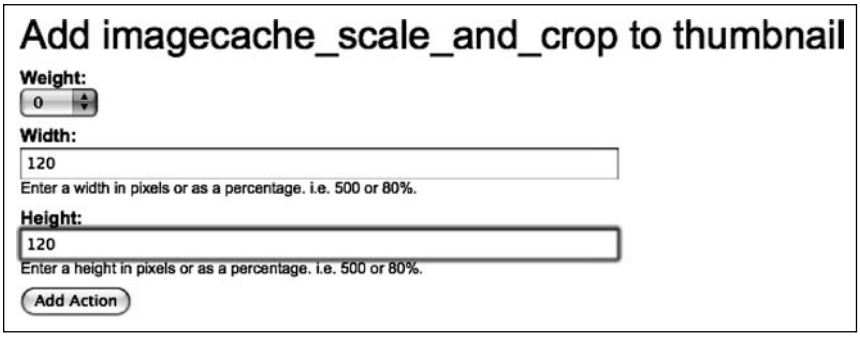


Рис. 7.21. *Настройки действий для ImageCache*

4. На этом создание набора правил thumbnail (миниатюра) можно считать законченным. Вернитесь на главную страницу настройки модуля ImageCache: Administer (Администрирование)→Site building (Конструкция сайта)→ImageCache (*admin/build/imagecache*) и щелкните на вкладке Add new preset (Добавить новый набор правил). Введите название preview (предпросмотр) в поле Preset Namespace (Название набора правил) и щелкните на кнопке Create New Preset (Создать новый набор правил).
5. В настройках нового набора правил в разделе New Actions (Новые действия) щелкните на ссылке Add Scale (Добавить масштабирование). Установите параметры операции масштабирования в соответствии с табл. 7.9, чтобы ограничить максимальную ширину изображения 480 пикселами. По окончании щелкните на кнопке Add Action (Добавить действие).

Таблица 7.9. Настройки операции масштабирования для набора правил «preview» (предварительный просмотр)

Параметр	Значение
Weight (Вес)	0
Width (Ширина)	480
Height (Высота)	*Оставить пустым*
Allow upscaling (Разрешить масштабирование вверх)	не отмечен

Теперь у нас имеется два законченных набора правил! Первый набор, «thumbnail» (миниатюра), используется для вывода миниатюр в списках из множества изображений, а второй, «preview» (предварительный просмотр), – для отображения изображений на страницах фотографий.

Настройка отображения поля Photo (Фотография)

Хотя мы уже создали два набора правил, наши фотографии по-прежнему отображаются в полный размер. Чтобы задействовать новые возможности отображения, нам необходимо настроить тип содержимого Photo (Фотография):

1. Перейдите на страницу настройки отображения для типа содержимого Photo (Фотография): Administer (Администрирование)→Content management (Содержание)→Content types (Типы содержимого) (*admin/content/types*) и щелкните на ссылке edit (изменить) для типа Photo (Фотография). На странице типа содержимого щелкните на вкладке Display fields (Поля) (*admin/content/node-type/photo/display*).
2. Измените настройки поля Photo (Фотография) в соответствии с табл. 7.10.

Таблица 7.10. Настройки отображения для поля Photo (Фотография)

Параметр	Значение
Label (Название)	<Hidden> (Скрыто)
Teaser (Анонс)	thumbnail image linked to node (изображение миниатюры со ссылкой на узел)
Full node (Весь материал)	preview image linked to image (изображение предварительного просмотра со ссылкой на изображение)

Теперь после настройки типа содержимого Photo (Фотография) и параметров отображения для использования ImageCache можно прерваться и попытаться выгрузить фотографии. После выгрузки в галерею не-

скольких изображений вы получите домашнюю страницу, как показано на рис. 7.22.

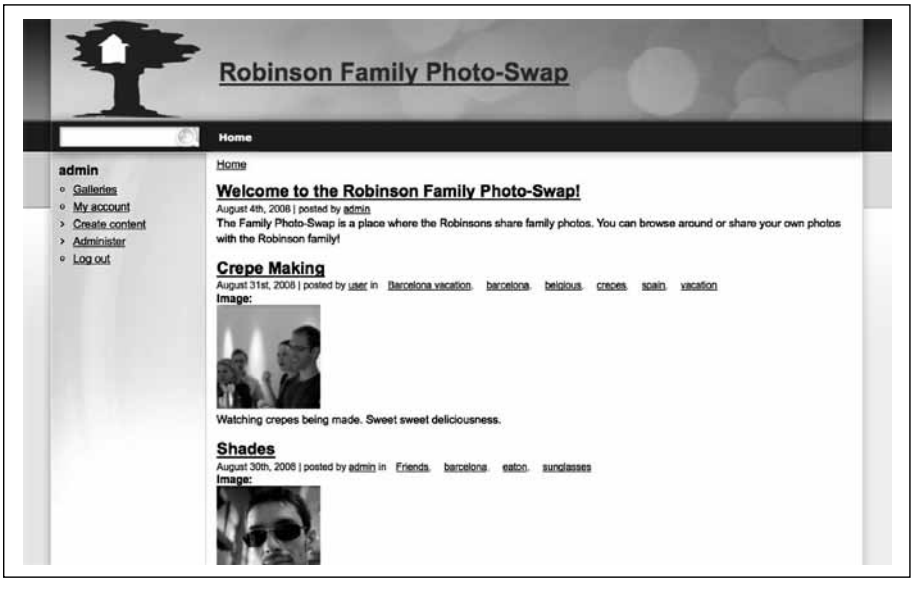


Рис. 7.22. *Главная страница сайта обмена фотографиями после создания набора правил для отображения миниатюр*

Улучшение качества изображений

Если сейчас взглянуть на миниатюры, сгенерированные модулем ImageCache, можно заметить, что качество изображений оставляет желать лучшего и сами изображения получились слишком сжатыми. При обработке изображений в формате JPEG модуль ImageCache использует параметр качества модуля ImageAPI, который по умолчанию имеет значение 75%. Увеличив это значение, можно получить более качественные изображения.

Для этого нужно выполнить следующие действия:

1. Перейдите на страницу настройки ImageAPI: Administer (Администрирование)→Site configuration (Настройка сайта)→ImageAPI (*admin/settings/imageapi*). Щелкните на вкладке Configure (Настроить).
2. Установите параметр JPEG quality (Качество JPEG) в значение 90% или выше и щелкните на кнопке Save configuration (Сохранить конфигурацию)
3. Чтобы увидеть действие этого параметра на существующие миниатюры, перейдите на страницу администрирования модуля ImageCache Administer (Администрирование)→Site building (Конструкция сайта)→ ImageCache (*admin/build/imagecache*).

4. Для каждого набора правил щелкните на ссылке Flush (Очистить), чтобы удалить старые, низкокачественные изображения. После этого при следующей попытке отображения будут сгенерированы новые изображения.

Помимо изображений, создаваемых модулем ImageCache, система Drupal также создает миниатюры в других местах, например аватары пользователей. При создании этих миниатюр настройка качества JPEG выполняется в другом месте, на странице Administer (Администрирование)→Site Configuration (Настройка сайта)→Image toolkit (Изображения) (*admin/settings/imagetoolkit*).

Практика: представление gallery

Теперь у нас имеется практически законченная процедура выгрузки изображений на сайт и отображение отдельных фотографий выглядит совсем неплохо. Однако сейчас миниатюры отображаются на отдельных страницах. Было бы предпочтительнее, если бы изображения располагались в виде сетки, что позволило бы показать набор изображений в пределах небольшой области. Для этого с помощью модуля Views можно создать представление. Когда мы закончим этот раздел, представление будет выглядеть, как показано на рис. 7.23.

Рис. 7.23. *Законченное представление gallery (галерея)*

1. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Modules (Модули) (*admin/build/modules*) и включите следующие модули:
   - Other (Другой)
     – Advanced Help

- Views
  - Views
  - Views UI

2. Перейдите на страницу администрирования модуля Views: Administer (Администрирование)→Site building (Конструкция сайта)→Views (*admin/build/views*). Щелкните на вкладке Add (Добавить) в верхней части страницы, чтобы создать новое представление. Установите настройки нового представления в соответствии с табл. 7.11.

Таблица 7.11. Базовые настройки для представления gallery (галерея)

Параметр	**Значение**
View name (Название представления)	gallery (галерея)
View description (Описание представления)	A gallery display of images (Галерея изображений)
View type (Тип представления)	Node (Материал)

3. Установите базовые настройки в разделе Page settings (Параметры страницы) в соответствии с табл. 7.12.

Таблица 7.12. Параметры страницы для представления gallery (галерея)

Defaults: Basic settings (базовые настройки)	**Значение**
Title (Название)	Galleries (Галереи)
Style (Стиль)	Style (Стиль): Grid (Сетка) Number of columns (Число столбцов): 4 Alignment (Выравнивание): Vertical (По вертикали)
Use pager (Использовать компонент выбора страниц)	Full pager (Полнофункциональный компонент)
Items per page (Элементов на страницу)	20
More link (Ссылка «далее»)	Create more link (Создавать ссылку «далее»): отмечено
Empty text (Текст для пустого представления)	No photos yet! (Пока нет фотографий!)

4. Найдите раздел Fields (Поля) представления, щелкните на ярлыке с изображением знака плюс, отметьте флажок Content: Image: Image (field_photo) (Содержимое: Изображение: Изображение (field_photo))

и щелкните на кнопке Add (Добавить). Выполните настройки в соответствии с табл. 7.13 и щелкните на кнопке Update (Обновить).

Таблица 7.13. Поля по умолчанию для представления gallery (галерея)

Defaults: Fields (По умолчанию: Поля)	Значение
Content: Image: Image (field_photo) (Содержание: Изображение: Изображение (field_photo))	Format (Формат): thumbnail image linked to node (изображение миниатюры со ссылкой на узел) Название: нет

5. Так как мы хотим отображать в наших списках только опубликованные изображения, добавим в представление несколько фильтров. Щелкните на ярлыке с изображением знака плюс в разделе Filters (Фильтры), отметьте перечисленные ниже фильтры и щелкните на кнопке Add (Добавить):
   - Node (Узел): Published (Опубликовано)
   - Node (Узел): Type (Тип)
   - Taxonomy (Таксономия): Vocabulary (Словарь)
6. Выполните настройки в соответствии с табл. 7.14, щелкая на кнопке Update (Обновить) для перемещения между формами с параметрами настройки.

Таблица 7.14. Фильтры по умолчанию для представления gallery (галерея)

Defaults: Filters (По умолчанию: Фильтры)	Значение
Node: Published (Узел: Опубликовано)	Published (Опубликовано): отмечено
Node: Type (Узел: тип)	Operator (оператор): Is one of «Photo» (Один из «Photo»)
Taxonomy: Vocabulary (Таксономия: Словарь)	Operator (оператор): Is one of «Gallery» (Один из «Gallery»)

7. Чтобы задать определенный порядок расположения фотографий, условимся первыми отображать самые последние полученные фотографии. Щелкните на ярлыке с изображением знака плюс в разделе Sort criteria (Критерий сортировки) и добавьте критерий Node: Post date (Узел: Дата создания).
8. По окончании настройки представления должны выглядеть, как показано на рис. 7.24. Щелкните на кнопке Save (Сохранить), чтобы сохранить представление.

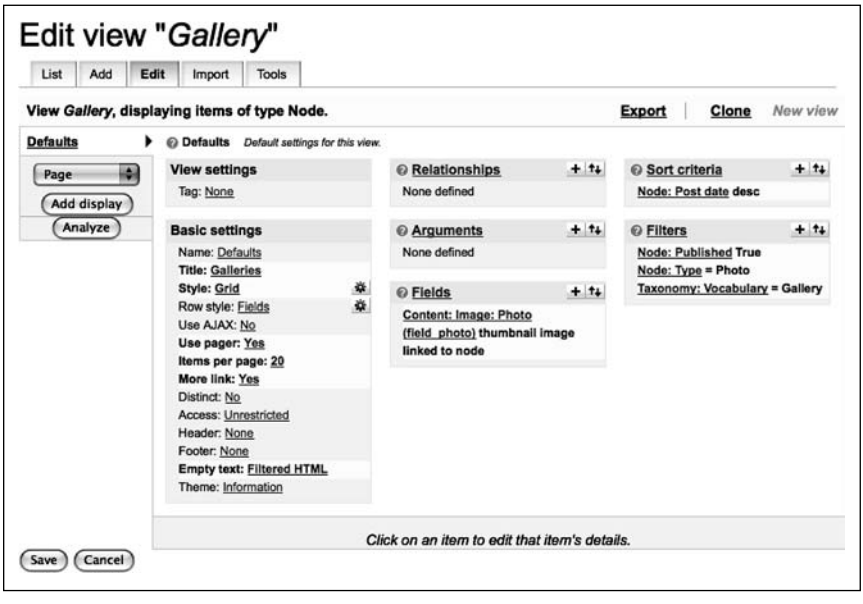


Рис. 7.24. *Настройки представления gallery (галерея)*

На этом мы закончили настройку отображения по умолчанию для нашего представления gallery (галерея). Щелкните на кнопке Live Preview (Оперативный просмотр) внизу страницы с настройками представления. В результате вы должны увидеть страницу, как показано на рис. 7.25.

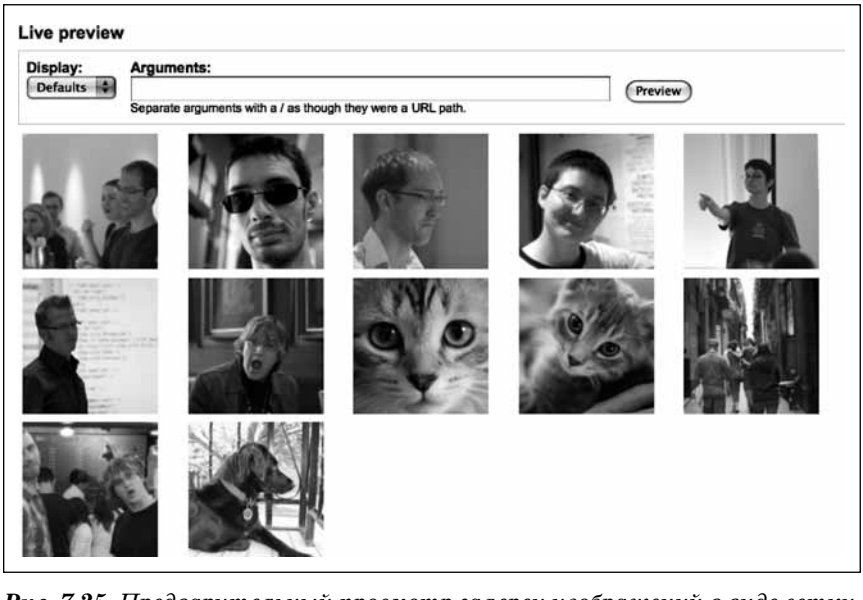


Рис. 7.25. *Предварительный просмотр галереи изображений в виде сетки*

Однако это представление пока нигде не отображается, и фотографии пока не классифицируются по именам галерей. Чтобы решить обе эти задачи, мы добавим в представление новое отображение:

1. Если вы уже покинули страницу с настройками представления, вернитесь на нее: Administer (Администрирование)→Site building (Конструкция сайта)→Views (admin/build/views) и щелкните на ссылке Edit (Изменить) для представления «gallery».
2. В списке отображений на странице слева выберите тип Page (Страница) и щелкните на кнопке Add display (Добавить отображение).
3. Назначьте новой странице адрес URL и элемент меню, настроив параметры нового отображения. Найдите раздел Page settings (Параметры страницы) и введите настройки в соответствии с табл. 7.15.

Таблица 7.15. Параметры страницы для представления gallery (галерея)

Page: Page settings (Страница: Параметры страницы)	**Значение**
Path (Путь)	gallery (галерея)
Menu (Меню)	Type (Еип): Normal menu entry (Обычный элемент меню) Title (Заголовок): Galleries (Галереи)

4. Чтобы отобразить изображения в определенной галерее, мы будем передавать представлению аргумент. При этом нам не нужно, чтобы представление включало этот аргумент по умолчанию (например, когда представление используется как блок). Щелкните на ярлыке с изображением знака плюс в разделе Arguments (Аргументы), отметьте флажок Taxonomy Term ID (Термин таксономии) и щелкните на кнопке Save (Сохранить). Установите параметры нового аргумента в соответствии с табл. 7.16 и щелкните на кнопке Update (Обновить).

Таблица 7.16. Настройки аргумента для представления gallery (галерея)

Page: Arguments (Страница: Аргументы)	**Значение**
Taxonomy (Таксономия): Term ID (Термин)	Title (Заголовок): %1 Action to take if argument is not present (Действие, выполняемое в случае отсутствия аргумента): Summary, sorted ascending (Сводный список, отсортированный по возрастанию) Validator (Валидатор): Taxonomy term (Термин таксономии) Vocabularies (Словари): Gallery

	Set the breadcrumb for the term parents (Устанавливать цепочку навигации для родительских терминов): отмечено

5. Вам будет предложено определить различные параметры. Щелкните на кнопке Update (Обновить), чтобы принять значения по умолчанию.
6. По окончании страница с настройками представления должна выглядеть, как показано на рис. 7.26. Обратите внимание, что настройки, унаследованные от отображения Defaults (По умолчанию), выводятся курсивом и имеют серый цвет. Значения, переопределенные для отображения Page (Страница), выводятся обычным шрифтом.
7. Щелкните на кнопке Save (Сохранить) в нижней части формы, чтобы сохранить представление.

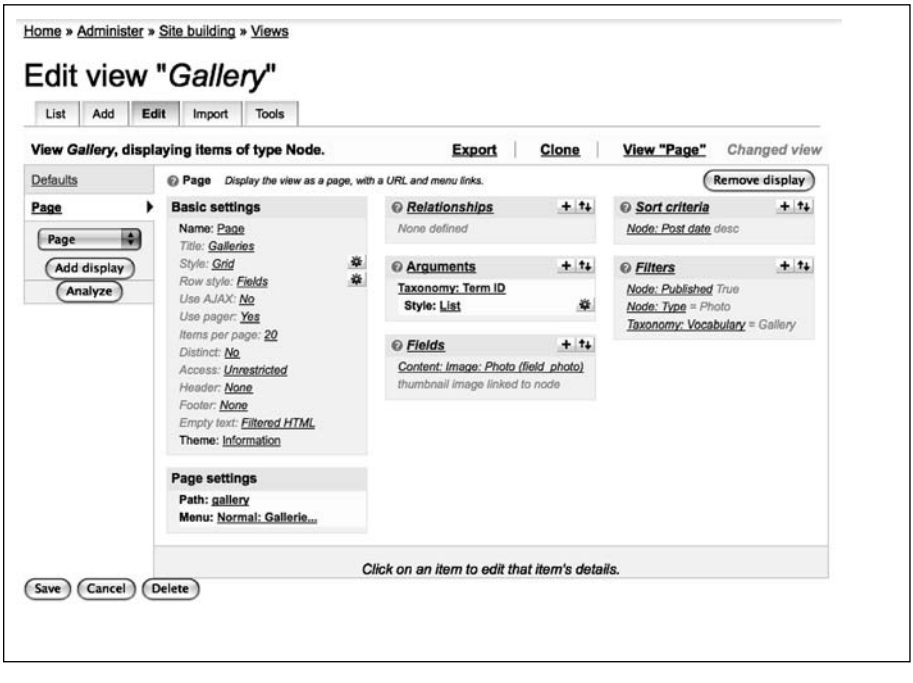


Рис. 7.26. *Настройки отображения Page (Страница) для представления gallery (галерея)*

После того как представление будет сконструировано, вы должны увидеть новый пункт Galleries (Галереи) в навигационном меню. Перейдите на страницу по этой ссылке, чтобы увидеть, как теперь выглядит наша новая фотогалерея! Обязательно выгрузите на сервер хотя бы несколько фотографий, и вы получите страницу, которая выглядит, как показано на рис. 7.27.

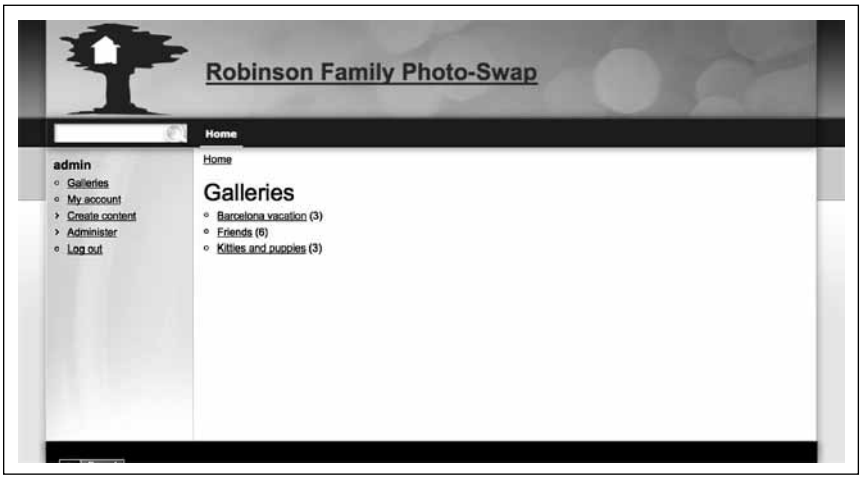


Рис. 7.27. *Сводное представление нашей галереи*

Это «сводная» версия нашей галереи. Она отображается при непосредственном вводе адреса URL, без дополнительных аргументов. Щелчок мышью на одной из галерей в списке даст вам адрес URL *gallery/(term id)*, где (*term id*) – это рубрика в галерее, определенная в словаре Gallery (Галерея).

После выбора определенной галереи вы должны увидеть страницу, напоминающую рис. 7.23 в начале этого раздела. Заголовок страницы Barcelona vacation (Каникулы в Барселоне) извлекается из рубрики таксономии, просматриваемой в настоящий момент, через значение аргумента *%1*, определенное в табл. 7.16 с настройками для раздела Arguments (Аргументы) в представлении gallery (галерея).

Практика: блок с последними фотографиями

Теперь у нас имеется достаточно удобное представление для просмотра фотографий, но вид нашей главной страницы со списком изображений по умолчанию все еще далек от желаемого. Давайте добавим в нее еще одно специальное представление, отображающее последние фотографии, выгруженные на сайт. Чтобы ускорить процедуру создания, мы скопируем представление gallery (галерея) и затем просто изменим некоторые параметры настройки. По окончании главная страница сайта будет выглядеть, как показано на рис. 7.28.

Другой способ, помимо копирования, заключается в том, чтобы создать еще одно отображение внутри существующего представления gallery (галерея). Однако, из-за того что нам придется убрать фильтр и внести некоторые изменения в настройки поля изображения, чтобы задействовать другой набор правил Image-

Cache, нам потребуется переопределить достаточно много настроек представления, имеющих значения по умолчанию. В подобных ситуациях, когда число переопределяемых настроек превышает число унаследованных, лучше всего начать с создания нового представления вместо того, чтобы создавать дополнительные отображения в существующем представлении.

Рис. 7.28. *Главная страница с блоком «Последние фотографии»*

1. Перейдите на страницу управления представлениями: Administer (Администрирование)→Site building (Конструкция сайта)→Views (*admin/build/views*).
2. В строке с названием представления щелкните на ссылке Clone (Копировать). В результате будет создана копия представления gallery (галерея), где достаточно будет выполнить всего несколько изменений.
3. Установите значения параметров в соответствии с табл. 7.17.

Таблица 7.17. Базовые настройки для представления latest_photos

Параметр	**Значение**
View name (Название представления)	latest_photo (последние фотографии)
View description (Описание представления)	A list of the latest photos on the site (Список последних фотографий на сайте)

4. Щелкните на кнопке Next (Далее), чтобы приступить к редактированию копии представления. Нам больше не нужно представление Page (Страница). Поэтому щелкните на вкладке Page (Страница) слева, а затем на кнопке Remove display (Удалить отображение). Щелкните на кнопке Save (Сохранить), чтобы полностью удалить отображение.
5. Точно так же нам больше не нужен фильтр Taxonomy: Vocabulary (Таксономия: Словарь). Щелкните на ссылке Taxonomy: Vocabulary (Таксономия: Словарь) в разделе Filters (Фильтры). В форме с настройками фильтра щелкните на кнопке Remove (Удалить).
6. В разделе Basic settings (Базовые настройки) представления установите параметр Title (Заголовок) в значение «Latest Photos» (Последние фотографии).
7. Добавьте в представление новое отображение Block (Блок). Выберите вкладку Block (Блок) в списке типов отображений слева и щелкните на кнопке Add display (Добавить отображение).
8. Найдите раздел Block settings (Параметры блока) для нашего нового отображения и установите параметр Admin в значение «Latest Photos» (Последние фотографии). В результате в административном разделе у этого блока появится соответствующий заголовок.
9. Щелкните на кнопке Save (Сохранить), чтобы обновить страницу с настройками нового представления latest_photos (последние фотографии), которая теперь должна выглядеть, как показано на рис. 7.29.

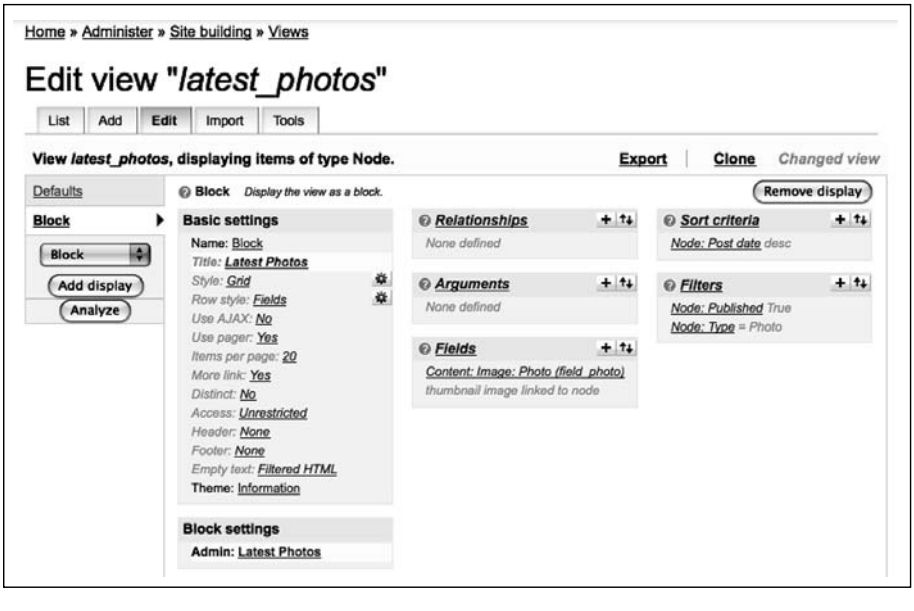


Рис. 7.29. *Страница с настройками блока Latest Photos (Последние фотографии)*

Это новое представление содержит блок, который можно использовать на любой странице сайта. В данном случае мы будем отображать его только на главной странице сайта, под текстом приветствия.

1. Перейдите на страницу настройки информации о сайте: Administer (Администрирование)→Site configuration (Настройка сайта)→Site information (О сайте) (*admin/settings/site-information*).
2. Измените значение *node* параметра Default front page (Главная страница по умолчанию) на значение *node/1* и щелкните на кнопке Save configuration (Сохранить конфигурацию). В результате этого шага на главной странице будет отображаться только содержимое узла 1 (страница с текстом «Welcome to the Robinson Family Photo-Swap» (Добро пожаловать на сайт обмена фотографии семьи Робинсон)). Теперь главная страница больше не будет отображать список узлов, как ранее.
3. Перейдите на страницу администрирования блоков Administer (Администрирование)→Site building (Конструкция сайта)→Blocks (Блоки) (*admin/build/block*).
4. Отбуксируйте мышью блок Latest Photos (Последние фотографии) в область Content bottom (Содержимое внизу). Щелкните на кнопке Save block (Сохранить блок), чтобы сохранить изменения.
5. Вы должны заметить, что теперь блок Latest Photos (Последние фотографии) присутствует внизу каждой страницы на сайте, включая страницу административного раздела, на которой мы сейчас находимся. Теперь мы сделаем так, чтобы блок появлялся только на главной странице. Щелкните на ссылке configure рядом с названием блока Latest Photos (Последние фотографии) и настройте блок в соответствии с табл. 7.18. По окончании щелкните на кнопке Save block (Сохранить блок).

Таблица 7.18. Настройки блока Latest Photos (Последние фотографии)

Параметр	**Значение**
Page-specific visibility settings (Установки видимости для страницы)	
Show block on specific pages (Показывать блок на определенных страницах)	Show on only the listed pages. (Отображать только на перечисленных страницах)
Pages (Страницы)	<front>

Теперь главная страница сайта The Robinson Family Photo-Swap выглядит просто замечательно. У нас реализовано создание уменьшенных копий двух различных размеров, отображаемых в привлекательном формате галереи, и мы можем помещать фотографии в блоки, чтобы отобра-

жать их на любой странице сайта, например, блок Latest Photos (Последние фотографии) на главной странице сайта.

Практика: модуль Custom Pagers

Однако остается еще одно место, где ощущается нехватка нашей любви и заботы, – это страница просмотра фотографий. При просмотре нескольких фотографий пользователь вынужден щелкать на фотографии, затем возвращаться на предыдущую страницу и щелкать на следующей фотографии. Этот процесс можно немного упростить, если добавить на страницу просмотра фотографии кнопки для перехода к предыдущей и следующей фотографии. Модуль Custom Pagers (*http://drupal.org/project/custom_pagers*) может принимать любое представление (такое, как наше представление gallery (галерея)) и создавать простой компонент постраничного просмотра, обеспечивающий возможность перехода к предыдущему и следующему элементу в представлении.

На рис. 7.30 показано, как будет выглядеть страница просмотра фотографии к концу этого раздела.

Наш компонент листания позволяет выполнять циклический обход фотографий, находящихся в одной галерее

Рис. 7.30. *Страница просмотра фотографий, теперь с возможностью листания*

1. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Modules (Модули) (*admin/build/modules*), включите следующие модули и щелкните на кнопке Save configuration (Сохранить конфигурацию):
   - Other (Другой)
     - Custom Pagers
     - Token
2. Перейдите на страницу администрирования модуля Custom pagers Administer (Администрирование)→Site building (Конструкция сайта)→ Custom pagers (*admin/build/custom_pagers*).
3. Щелкните на ссылке Add a new custom pager (Добавить новый компонент постраничного просмотра) внизу страницы.
4. Заполните форму значениями из табл. 7.19, как показано на рис. 7.31, и щелкните на кнопке Submit (Отправить).

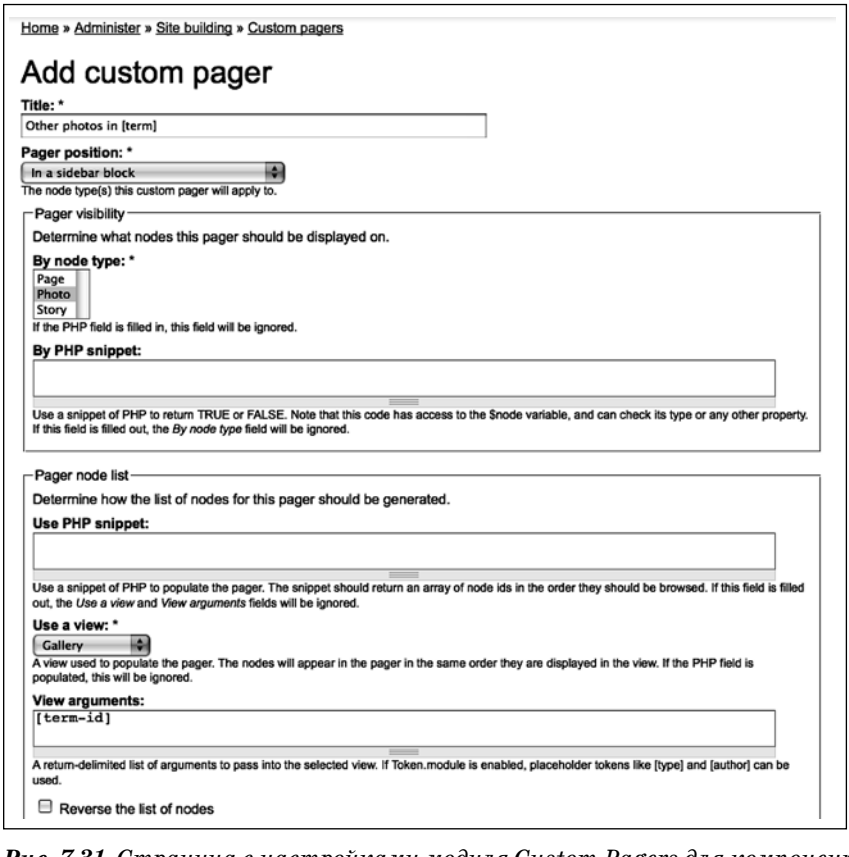


Рис. 7.31. *Страница с настройками модуля Custom Pagers для компонента постраничного просмотра галереи*

Таблица 7.19. Настройки модуля Custom Pager для организации навигации по фотографиям

Параметр	Значение
Title (Заголовок)	Other photos in [term] (Другие фотографии в галерее [term])
Pager position (Местоположение)	In a sidebar block (В блоке боковой колонки)
Pager visibility (Видимость)	
By node type (По типу узла)	Photo
Pager node list (Список узлов)	
Use a view (Использовать представление)	gallery
View arguments (Аргументы представления)	[term-id]

В качестве значений лексем [*term*] и [*term-id*] всегда используется **самый верхний** термин, связанный с узлом. Мы определили вес словаря Gallery, как –1 (а вес словаря Tags, как 0), поэтому лексемы с названием рубрики и идентификационным номером текущей галереи будут заполняться динамически.

5. Теперь, когда у нас имеется настроенный компонент постраничного просмотра, готовый к отображению на странице в виде блока, перейдите на страницу администрирования блоков Administer (Администрирование)→Site building (Конструкция сайта)→Blocks (Блоки) (*admin/build/blocks*).
6. Найдите блок «Other photos in [term]» (Другие фотографии в галерее [term]) и отбуксируйте его в область Left sidebar (Левая колонка).
7. Щелкните на кнопке Save blocks (Сохранить блоки), чтобы сохранить изменения.

Теперь перейдите на страницу просмотра фотографий. В левой колонке должны появиться удобные ссылки для перехода к предыдущей и следующей фотографии в той же галерее, что и изображение, просматриваемое в настоящий момент.

Дополнительно

Если к настоящему моменту вы построили галерею, значит, вы создали настоящую галерею изображений на базе платформы Drupal! Но не стоит останавливаться на достигнутом. Самое важное в создании сайтов

с помощью Drupal заключается в том, что вы постоянно можете улучшать их, добавляя новые возможности к самым разным частям сайта. В этом разделе описываются некоторые дополнительные функциональные возможности, часто встречающиеся на веб-сайтах, где пользователи обмениваются фотографиями:

Community Tags (http://drupal.org/project/community_tags)

Маркировка членами сообщества – это практически то же самое, что и произвольная маркировка, за исключением того, что в этом случае возможность добавлять теги в содержимое имеет любой посетитель (с соответствующими разрешениями), а не только автор. Обеспечить такой способ маркировки легко можно с помощью модуля Community Tags.

Tagadelic (http://drupal.org/project/tagadelic)

Маркировка членами сообщества и облака тегов – это две особенности, которые часто используются вместе. Модуль Tagadelic делает создание облаков тегов в Drupal очень простым делом. Он предоставляет блоки, которые могут отображать популярные термины из различных словарей, подчеркивая их вес. Наиболее популярные термины выводятся в облаке более крупным шрифтом, как показано на рис. 7.32.

Рис. 7.32. *Пример облака тегов, реализованного с помощью модуля Tagadelic*

Taxonomy Redirect (http://drupal.org/project/taxonomy_redirect)

В настоящее время страницы галерей на сайте Robinson Family Photo-Swap прекрасно оформлены наборами миниатюр, но страницы с рубриками таксономии по-прежнему отображают простые списки узлов. Это означает, что все то, что мы вложили в представление gallery, мы не можем показать, когда пользователь щелкнет на названии рубрики под фотографией или в облаке тегов.

Чтобы ликвидировать этот недостаток, можно использовать модуль Taxonomy Redirect, который создаст ссылки, указывающие на новое местоположение в галерее, а не на адрес *taxonomy/term*. Благодаря этому все ссылки на рубрики в обоих словарях будут вести к представлению с миниатюрами – вместо страницы, предоставляемой модулем Taxonomy по умолчанию.

В заключение

В этой главе мы познакомились с модулями ImageCache, ImageField и Custom Pagers. Мы сконструировали тип содержимого Photo (Фотография) и настроили его отображение через комбинацию наборов правил модуля ImageCache, представлений и собственного компонента постраничного просмотра. У нас имеется возможность добавить маркировку членами сообщества, облака тегов и сделать так, чтобы все ссылки на рубрики вели к представлению gallery вместо простого списка рубрик, используемого по умолчанию.

Ниже приводится перечень модулей, упоминавшихся в этой главе:

- Community Tags: *http://drupal.org/project/community_tags*
- Content Construction Kit (CCK): *http://drupal.org/project/cck*
- Custom Pagers: *http://drupal.org/project/custom_pagers*
- ImageCache: *http://drupal.org/project/imagecache*
- ImageCache Actions: *http://drupal.org/project/imagecache_actions*
- ImageField: *http://drupal.org/project/imagefield*
- Tagadelic: *http://drupal.org/project/tagadelic*
- Taxonomy Redirect: *http://drupal.org/project/taxonomy_redirect*
- Token: *http://drupal.org/project/token*
- Views: *http://drupal.org/project/views*

Ниже приводится список ресурсов, упоминавшихся в этой главе, и ресурсов сообщества для более подробного изучения новых концепций, представленных в этой главе:

- Все модули, имеющие отношение к пакету CCK: *http://drupal.org/project/Modules/category/88*
- Поддержка функции Clean URL: *http://drupal.org/node/15365*
- Рабочая группа Image: *http://groups.drupal.org/image*

8

Многоязычные сайты

Создание веб-сайтов, информационное наполнение для которых создается сообществом, – это замечательно, но как быть, если не все или никто в вашем сообществе не говорит по-английски? Мир велик, и лишь 6% людей считают английский своим родным языком. Многоязычные сайты позволят обращаться к членам вашего сообщества и обеспечат для них возможность привносить свое информационное наполнение. Поддержка нескольких языков на сайте – это не просто возможность для пользователей отправлять свои материалы на том языке, на каком они пожелают. Существует еще ряд проблем, которые необходимо принимать во внимание, такие как навигация по сайту, форматирование дат и вывод справочного текста. А как быть с возможностью легко переключаться между версиями одного и того же сообщения, написанными на разных языках? Стоит только начать задумываться обо всех этих проблемах, как они начинают расти и множиться. К счастью, ядро системы Drupal и несколько сторонних модулей уже берут на себя большую часть этой работы, благодаря чему мы можем сконцентрироваться на создании нашего сообщества и информационного наполнения.

Создание многоязычных сайтов неразрывно связано с двумя крупными понятиями: интернационализация (internationalization), часто обозначается, как i18n, и локализация (localization), часто обозначается, как l10n. *Интернационализация* – это несущая структура, позволяющая программному обеспечению адаптироваться к различным языкам, а *локализация* – это процесс фактического перевода программного обеспечения для использования в определенных регионах. Локализация не ограничивается только переводом текста – она также затрагивает такие вещи, как форматы представления дат и валюты.

Версия Drupal 6 достигла значительных успехов в реализации системы интернационализации внутри ядра Drupal, что обеспечивает значительную простоту локализации. В действительности ядро предоставля-

ет не все инструменты, необходимые для полной локализации сайтов, но существуют сторонние модули, готовые восполнить этот недостаток.

В этой главе будут представлены следующие модули:

Locale (базовый)

Реализует интерфейс для перевода и импортирования переводов элементов пользовательского интерфейса.

Content Translation (базовый)

Обеспечивает перевод содержимого, созданного пользователем.

Internationalization (http://drupal.org/project/i18n)

Обеспечивает возможность перевода других элементов, таких как меню, блоки и рубрики таксономии.

Localization Client (http://drupal.org/project/l10n_client)

Простой в использовании интерфейс к модулю Locale.

Book (базовый)

Модуль, позволяющий нескольким пользователям совместно работать над созданием документации.

Forum (базовый)

Простая система обмена мнениями по темам.

Рис. 8.1. *Законченный веб-сайт Migratory Patterns*

Если вы собираетесь следовать за практическими упражнениями этой главы, вам следует установить Drupal, используя профиль установки Multilingual из примеров кода, распространяемых вместе с книгой. В результате применения этого профиля на вашем веб-сервере будет создана заготовка веб-сайта. В окончательном виде этот веб-сайт будет выглядеть, как показано на рис. 8.1 и по адресу *http://multilingual.usingdrupal.com*. За дополнительной информацией об использовании профилей установки обращайтесь к разделу «Вступление».

Задача

Наш клиент, Migratory Patterns, – это международная группа, занимающаяся изучением перелетных птиц. Ей необходим веб-сайт, содержащий форум, куда мог бы прийти каждый желающий и принять участие в обсуждении интересующих его тем, а также база знаний, накопленных сообществом. Группе требуется, чтобы сайт имел форумы на разных языках и позволял бы ее членам просматривать содержимое сайта на их родных языках. Группе также необходимо хранилище общих знаний, с помощью которого ее члены могли бы делиться своей информацией, и способ отправки новой информации на сайт. Кроме того, сайт должен обеспечивать возможность перевода содержимого базы знаний членами группы, владеющими более чем одним языком, чтобы они могли совместно пользоваться накопленными знаниями.

В настоящее время в группе имеются члены, говорящие на трех языках – английском, голландском и французском, но им хотелось бы иметь возможность по мере развития группы добавлять другие языки.

Примечания к реализации

Для публикации новостей на главной странице легко может использоваться базовый тип содержимого Story (Статья), входящий в ядро Drupal, однако остальные требования клиента требуют дополнительного обсуждения.

Форумы

Существует несколько сторонних модулей, обеспечивающих интеграцию между Drupal и другими системами создания форумов, такими как phpBB. Однако в составе системы Drupal имеется собственная реализация форума в виде встроенного модуля Forum. Модуль Forum использует такие базовые концепции системы Drupal, как таксономия (для контейнеров форумов и самих форумов), узлы (для сообщений внутри форумов) и комментарии (для ответов), что обеспечивает бесшовную интеграцию с остальными частями веб-сайта, включая средства перевода.

База знаний

Другой модуль, встроенный в систему Drupal, модуль Book, обеспечивает возможность одновременной работы нескольких пользователей над созданием комплекта документации. Страницы содержимого типа Book page (Страница подшивки) могут образовывать одну или более иерархий с автоматическим созданием ссылок на предыдущую, следующую и начальную страницы на каждой из них. Кроме того, для каждой страницы имеется «версия для печати», которая включает неформатированное содержимое текущей страницы и любые подстраницы, что обеспечивает простоту печати или загрузки страниц на локальный компьютер для последующего чтения.

Перевод пользовательского интерфейса

Под пользовательским интерфейсом здесь подразумевается текст, содержащийся в системе Drupal, – как в ядре, так и в сторонних модулях. Сюда входят названия полей форм, справочный текст и текст в элементах навигации. Встроенный модуль Locale предоставляет платформу, позволяющую выполнять перевод пользовательского интерфейса. Чтобы получить большую часть нашей локализации, необходимо загрузить переводы пользовательского интерфейса, поставляемые нами. При добавлении сторонних модулей необходимо проверять наличие переводов для этих модулей. Не стоит отчаиваться, если таких переводов не существует, потому что ядро также обеспечивает возможность добавлять и обновлять переводы по мере необходимости. Модуль Localization Client использует эту платформу и реализует отличный дружественный интерфейс, упрощающий перевод элементов пользовательского интерфейса.

Однако модуль Locale не в состоянии охватить все аспекты пользовательского интерфейса, поэтому мы будем использовать отличный модуль Internationalization (i18n), восполняющий эти недостатки. Модуль Internationalization предоставляет в наше распоряжение инструменты для перевода рубрик таксономии, блоков и некоторых переменных, хранящих информацию о сайте, таких как название сайта и декларация о намерениях (миссия). Помимо предоставления дополнительного перевода этот модуль помогает управлять многоязычным содержимым.

Перевод содержимого, созданного пользователями

Последнее, чего нам не хватает, – это возможности перевода всего содержимого сайта, созданного пользователями: сообщений в форумах, страниц и тому подобного. В состав ядра системы входит модуль Content Translation, помогающий выполнять эту тяжелую работу. Он позволит нам решить, какое содержимое можно переводить, и даст возможность создать несколько версий одного и того же узла, по одной для каждого языка. Кроме того, он упрощает переключение между языками. И снова модуль Internationalization восполнит некоторые недостатки.

В центре внимания: средства интернационализации в ядре

В состав ядра Drupal входят два модуля, предназначенные для работы с разными языками: Locale и Content Translation. Модуль Locale работает с пользовательским интерфейсом и обеспечивает нас неплохим набором инструментов, позволяющим импортировать существующие переводы, создавать или изменять наши собственные и экспортировать переводы с нашего сайта для использования на других сайтах. Модуль Content Translation позволяет создавать несколько версий одного и того же содержимого на различных языках и связывать переведенные версии вместе.

Еще одной, по-настоящему замечательной особенностью ядра Drupal 6 является поддержка языков с письмом справа налево (right-to-left, RTL), таких как арабский или иврит. Если язык определяется как язык с письмом справа налево, система Drupal будет автоматически переворачивать весь текст, чтобы он читался в правильном направлении, как показано на рис. 8.2, где текст написан на иврите.

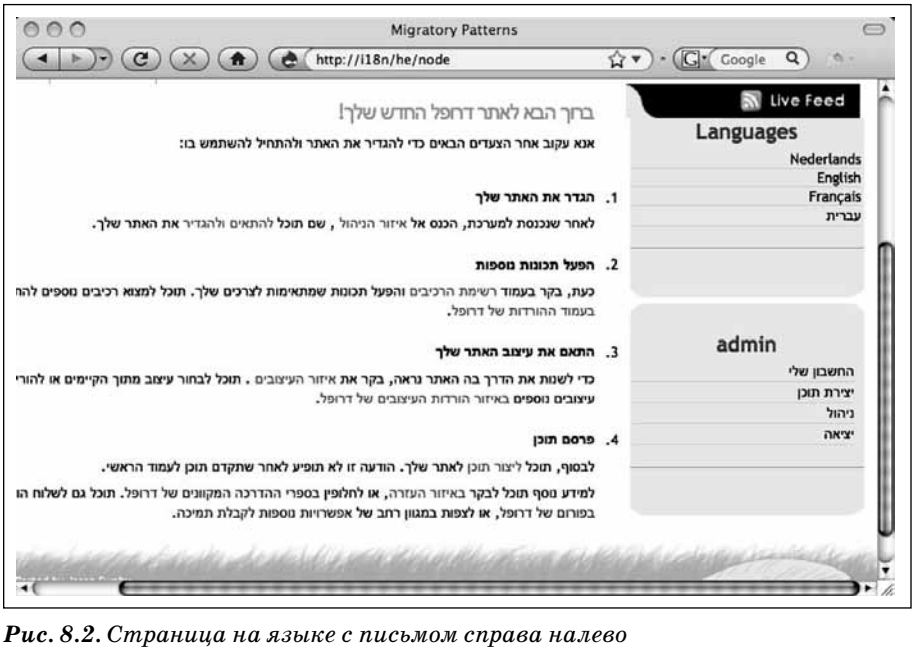


Рис. 8.2. *Страница на языке с письмом справа налево*

Модуль Locale

Модуль Locale обеспечивает механизмы переключения интерфейса сайта между файлами переводов и языками. Давайте подробнее рассмотрим, что это означает.

Переводы

«Перевод» – это обычный файл или коллекция файлов, содержащих информацию в стандартном формате. Имена файлов переводов, следующих этому формату, имеют расширение *.po*, которое является сокращением от Portable Object (переносимый объект). Файл с расширением *.po* – это обычный текстовый файл, который идентифицирует текстовые строки и содержит переводы этих строк на определенный язык. В системе Drupal файлы переводов содержат список всех текстовых строк, встречающихся в пользовательском интерфейсе, и их переведенные версии.

Проекты переводов Drupal, которые можно загрузить по адресу *http://drupal.org/project/translations*, – это специальным образом организованные и упакованные группы файлов *.po*, структура которых полностью совпадает со структурой каталогов Drupal. Эти проекты охватывают перевод только строк, входящих в состав ядра Drupal, то есть пользовательского интерфейса, содержащегося в ядре Drupal. Важно отметить, что эти проекты могут находиться на разных стадиях готовности, и иногда вы можете не согласиться с тем, как выполнен перевод той или иной строки. Не стоит волноваться, мы покажем, как исправить эту проблему.

В дополнение к проектам перевода модулей ядра файлы переводов также могут предоставляться сторонними модулями. В этом случае они могут содержаться внутри загружаемого архива модуля. Сопровождение модулей переводами – не самая тщательно выполняемая задача, а так как модули могут обновляться значительно чаще, чем ядро Drupal, имеющиеся переводы могут быстро устаревать.

Мы рассмотрим автоматическую загрузку файлов переводов, имеющуюся в Drupal, которая является стандартным способом доступа к ним. Кроме того, в случае необходимости имеется возможность вручную импортировать отдельные файлы *.po*.

Перевод интерфейса

Чтобы перевести интерфейс Drupal на многие языки, трудилось множество добровольцев. Вы можете обнаружить необходимость добавить или изменить используемый перевод. На этот случай в Drupal имеются инструменты, которые помогут вам.

Если перейти на страницу перевода интерфейса Administer (Администрирование)→Site building (Конструкция сайта)→Translate interface (Переводить интерфейс) (*admin/build/translate*), вы увидите список языков, а также количество и процент уже переведенных строк, как показано на рис. 8.3. При перемещении по сайту Drupal отслеживает все строки интерфейса, с которыми вы сталкиваетесь. Это возможно благодаря тому, что строки, допускающие возможность перевода, идентифицируются непосредственно в программном коде посредством применения функции перевода (функция `t()`). Как только открывается какая-либо

страница, все строки, допускающие возможность перевода, оказываются доступны для поиска и перевода. Открытие страницы является важным шагом, о котором легко забыть. Если вы начинаете поиск слов, которые заведомо присутствуют на сайте, но страницу, где они упоминаются, фактически не открывали, поиск по интерфейсу не увенчается успехом. Подробнее о переводе мы поговорим в разделе «Практика: перевод интерфейса» ниже в этой главе.

Translate interface

Overview | Search | Import | Export

This page provides an overview of available translatable strings. Drupal displays translatable strings in text groups; modules may define additional text groups containing other translatable strings. Because text groups provide a method of grouping related strings, they are often used to focus translation efforts on specific areas of the Drupal interface.

Review the languages page for more information on adding support for additional languages.

Language	Built-in interface
Dutch	2612/2636 (99.09%)
English (built-in)	n/a
French	2454/2636 (93.1%)

Рис. 8.3. *Количество и процент переведенных строк*

Важно помнить, что любой перевод, выполненный посредством интерфейса Drupal, а не импортированный в виде файла *.po*, будет сохраняться не в файле, а в базе данных. В системе Drupal имеется механизм экспортирования, который позволит вам сохранить перевод в виде файла, который затем можно будет использовать на других сайтах.

Если вам действительно придется поработать над переводом для вашего сайта, вам определенно следует рассмотреть возможность передачи результатов своего труда сообществу. Передавая свои переводы, вы поможете не только многочисленному сообществу Drupal, но и самому себе, так как найдется немало людей, которые протестируют ваш перевод и помогут сопровождать его.

Переключение языка

У каждого сайта должен быть язык по умолчанию, но как добиться отображения информации на других языках? Для этого существуют два основных механизма: настройка выбора языка и блок переключения языка. После того как будет установлен и включен хотя бы еще один язык, первое, что необходимо сделать, – это сообщить системе Drupal, как автоматически обслуживать несколько языков посредством настройки параметра выбора языка. По умолчанию система Drupal не делает ничего, и пользователи будут вынуждены вручную выбирать свой язык. У вас имеется несколько вариантов автоматического выбора языка на основе информации в адресе URL, приведенных на рис. 8.4.

Language negotiation:

- ○ None.
- ○ Path prefix only.
- ◉ Path prefix with language fallback.
- ○ Domain name only.

Select the mechanism used to determine your site's presentation language. **Modifying this setting may break all incoming URLs and should be used with caution in a production environment.**

Рис. 8.4. *Параметры автоматического переключения языка*

Система Drupal может выбирать язык на основе доменного имени сайта или на основе префикса. Если вы собираетесь использовать отдельные доменные имена для каждого языка, вы можете определить в настройках языка соответствия между доменными именами и языками. Например, можно настроить использование голландского языка для доменного имени *http://nl.example.com* или даже *http://foo.example.com*. Всякий раз, когда кто-либо будет обращаться к сайту с использованием этих доменных имен, будет использоваться язык, который вы определите.

Еще более типичный способ выбора языка заключается в использовании префикса пути. И снова вы можете настроить любые префиксы, какие только пожелаете. По умолчанию установленный перевод определяет соответствуеющий ему код языка как идентификатор префикса в пути. При такой настройке Drupal будет проверять путь на наличие кода языка непосредственно за доменным именем, например *http://example.com/nl/forum*. Если система Drupal обнаружит допустимый код, она будет отображать интерфейс на языке, ассоциированным с этим кодом. Префикс не требуется при использовании языка по умолчанию, поэтому для всех путей, не «осложненных» префиксом, будет использоваться этот язык, например *http://example.com/forum*. Существуют еще два варианта настройки, применяемые при использовании префиксов, которые позволяют указать, что делать, если префикс не найден. При выборе параметра Path prefix only (Префикс пути) будет выполняться поиск префикса в пути и в случае его отсутствия будет использоваться язык по умолчанию. При выборе параметра Path prefix with language fallback (Префикс пути с определением языка) система Drupal выполнит дополнительные проверки в других местах. При выполнении дополнительных проверок будет просмотрен профиль пользователя: возможно, пользователь указал предпочитаемый язык на странице My Account (Мой профиль). Если там ничего не будет найдено, будет проверена настройка языка в броузере и произведена попытка отыскать его среди доступных языков. Если все эти проверки не увенчаются успехом, то будет использоваться язык по умолчанию.

Настройки выбора языка определяют, как система Drupal будет автоматически изменять язык отображения, однако пользователи могут сами выбирать язык. После того как будет установлен и включен хотя бы еще

один язык, на странице My Account (Мой профиль) появится новый раздел Language settings (Языковые настройки) с простым списком радиокнопок, соответствующих доступным языкам, как показано на рис. 8.5. Однако эти языковые настройки доступны только авторизованным пользователям. Чтобы предоставить выбор всем пользователям, включая анонимных, можно включить встроенный блок переключения языка, в котором перечислены все доступные языки и который позволяет выбирать язык отображения информации.

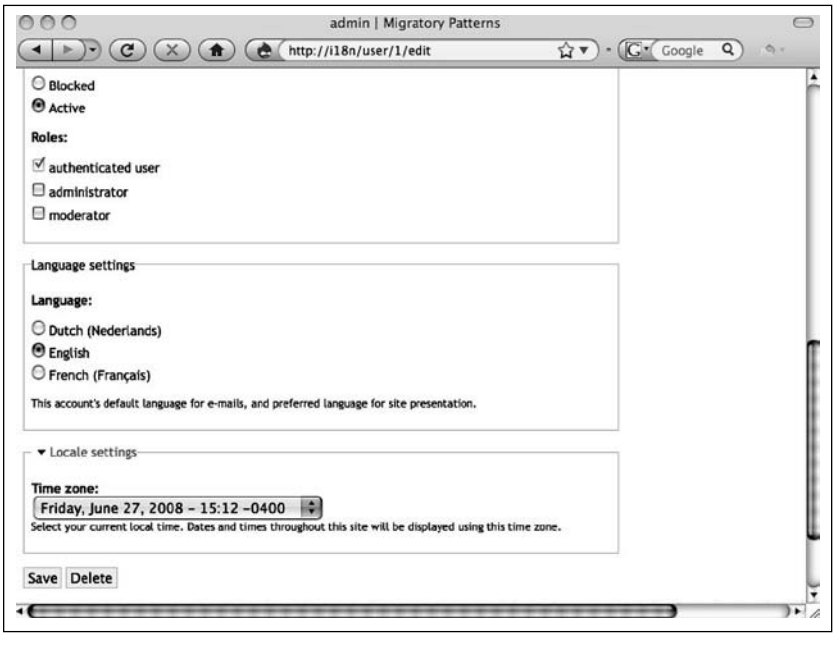


Рис. 8.5. *Настройка языка на странице My Account (Мой профиль)*

Локализованный инсталлятор

Добавлять новые переводы можно в любое время, однако, если перевод добавляется перед установкой Drupal, процесс установки может протекать на выбранном вами языке. По умолчанию на самой первой странице инсталлятор предоставляет на выбор два варианта: Install Drupal in English (Выполнить установку Drupal на английском языке) или Learn how to install Drupal in other languages (Узнайте, как установить Drupal на других языках). Если в наличии имеется один или более переводов, вместо этого будет предложена группа радиокнопок, соответствующих языкам, из которых вы можете выбирать, как показано на рис. 8.6. После этого все остальные страницы в процессе установки будут выводиться на выбранном языке. По окончании установки Drupal выбранный язык будет установлен как язык по умолчанию для сайта, и система Drupal включит модуль Locale.

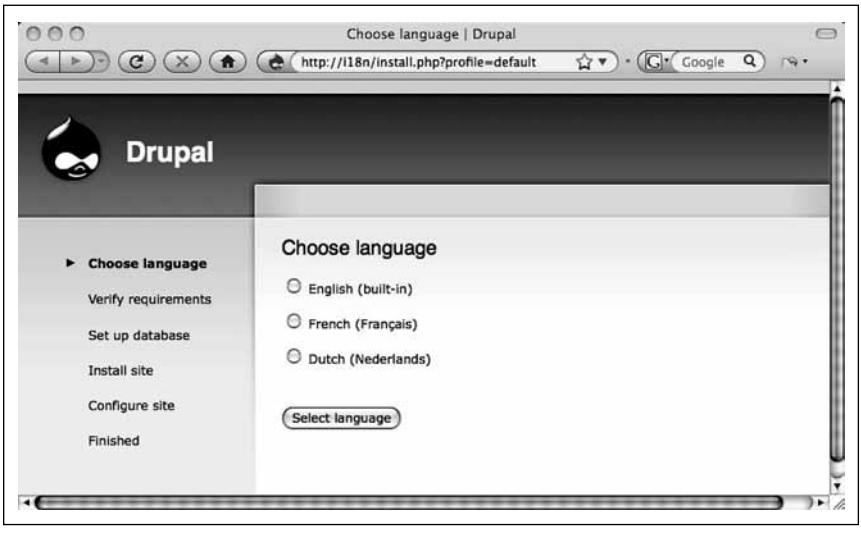


Рис. 8.6. *Инсталлятор предоставляет возможность выбора из доступных переводов*

Перевод содержимого

Когда дело доходит до перевода содержимого, Drupal интерпретирует каждый перевод как самостоятельный элемент информационного наполнения. Поддержку нескольких языков можно включить для каждого типа содержимого в отдельности, и при этом на выбор имеется два пути, в зависимости от того, предпочтете ли вы идентифицировать различные языки или создавать взаимосвязанные «версии» для каждого элемента содержимого.

Модуль Locale дает возможность обеспечить поддержку нескольких языков для типов содержимого. При включенной поддержке в форме ввода содержимого появится раскрывающийся список, как показано на рис. 8.7, позволяющий выбирать язык для создаваемого элемента содержимого. Это обеспечивает возможность идентификации языка, используемого для данного содержимого.

При использовании модуля Content Translation появляется дополнительный параметр поддержки нескольких языков: Enabled, with translation (Включено, с переводом). Использование этой настройки позволит не только идентифицировать язык создаваемого элемента содержимого, но и привязывать другие узлы, как переведенные версии данного содержимого. Например, можно создать на сайте страницу About (О сайте), которая будет иметь то же самое содержимое на французском и голландском языках. При включенной поддержке переводов можно было бы создать оригинальную страницу About (О сайте) на английском языке, а затем совершенно новые узлы для французской и голландской

версий. Система Drupal будет следить за этими тремя узлами и помнить, что они связаны друг с другом и что каждый из них является всего лишь версией одной и той же страницы. На каждой странице внизу под содержимым будут выводиться ссылки на другие переводы, как показано на рис. 8.8.

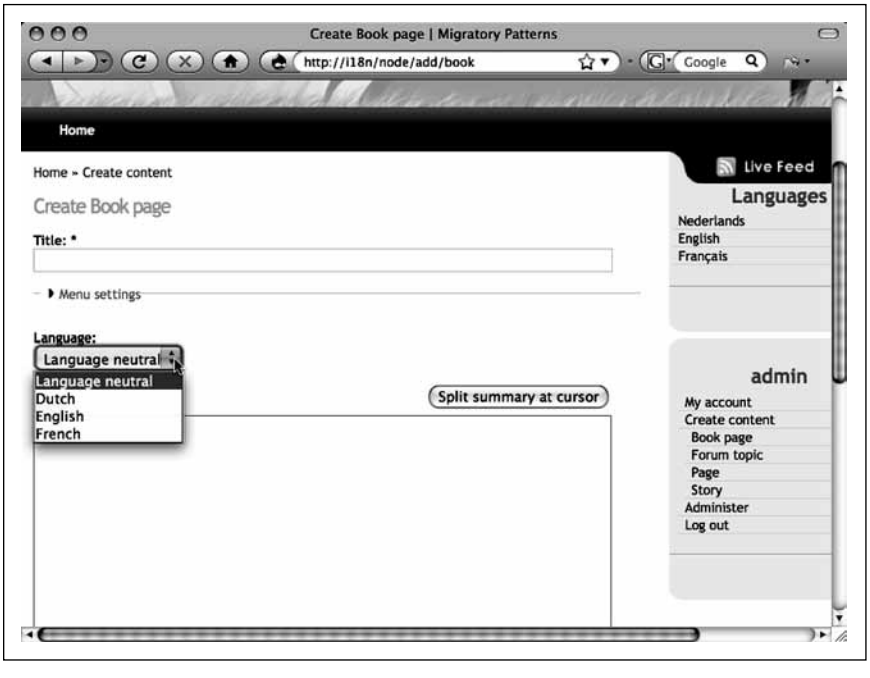


Рис. 8.7. *Выбор языка для нового элемента содержимого*

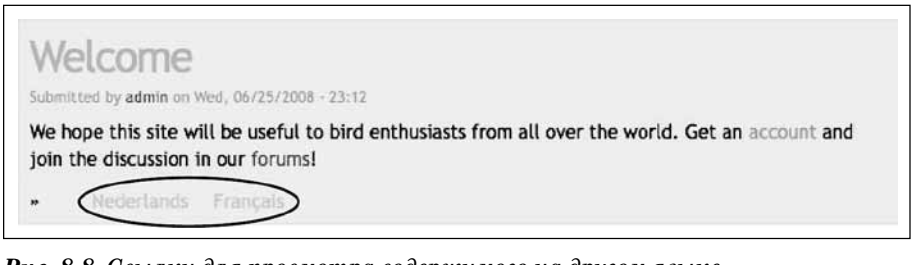


Рис. 8.8. *Ссылки для просмотра содержимого на другом языке*

Практика: установка перевода

Первый шаг к использованию поддержки нескольких языков в Drupal заключается в установке перевода, чтобы обеспечить возможность выбора из нескольких языков. Этот шаг включает два этапа: загрузка перевода и его извлечение и настройка модуля Locale на его распознавание:

1. Для начала следует загрузить перевод со страницы *http://drupal.org/project/translations*, например, *http://drupal.org/project/nl*. Как и в случае с модулями и темами оформления, загружаемая версия должна соответствовать версии Drupal.

Если вы опытный пользователь и обычно извлекаете исходные тексты Drupal из репозитория CVS, вы должны понимать, что в случае переводов требуется устанавливать загружаемые версии, а не извлекать их из CVS. В процессе упаковки файлов переводов в архиве создается корректная структура каталогов, а для работы с переводами, извлеченными из CVS, придется воссоздавать эту структуру вручную.

2. Распакуйте архив с переводами в корневой каталог Drupal (то есть в тот, где находится файл *index.php*), как показано на рис. 8.9.

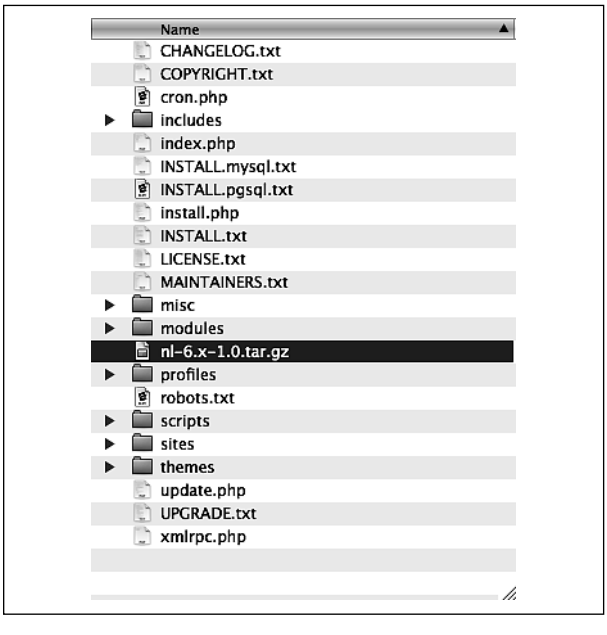


Рис. 8.9. *Архив с файлами перевода загружен в корневой каталог Drupal*

После распаковывания архива в корневом каталоге должны появиться новые файлы с кодом языка в имени (такие как *LICENSE.nl.txt*). Если в корневом каталоге Drupal появится новый каталог с кодом языка в имени (такой как *nl-6.x-1.0*), следовательно, установка прошла корректно. Чтобы убедиться, что файлы перевода находятся в нужном месте, можно заглянуть в каталог *profiles/default*, где вы должны увидеть новый каталог, с файлами переводов (такими как *nl.po*) внутри него, как показано на рис. 8.10.

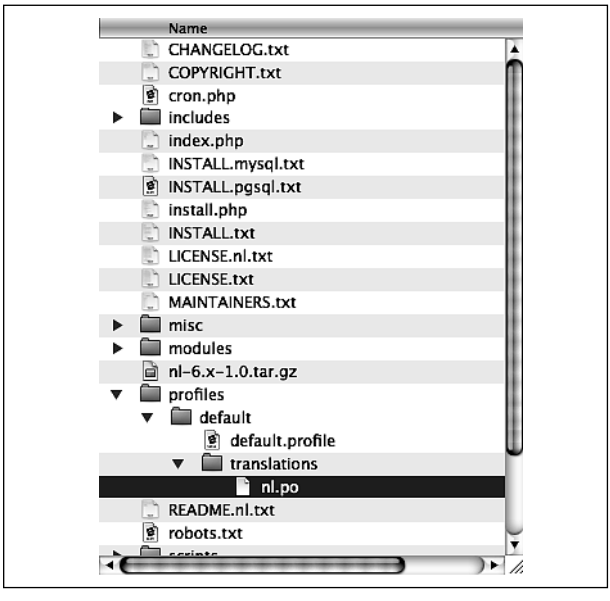


Рис. 8.10. *Содержимое каталога profile с новым файлом .po*

3. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Modules (Модули) (*admin/build/modules*), включите модуль Locale и щелкните на кнопке Save configuration (Сохранить конфигурацию). Обратите внимание: если вы устанавливали Drupal на языке, отличном от английского, этот шаг уже был выполнен автоматически в процессе установки.
4. Перейдите на страницу Administer (Администрирование)→User management (Управление пользователями)→Permissions (Разрешения) (*admin/user/permissions*) и настройте разрешения в соответствии с табл. 8.1. Сохраните изменения, щелкнув на кнопке Save permissions (Сохранить права доступа).

Таблица 8.1. Разрешения для модуля Locale

Разрешение: модуль locale	**anonymous user (анонимный пользователь)**	**authenticated user (зарегистрированный пользователь)**	**editor (редактор)**	**site administrator (администратор сайта)**
administer languages (управлять языками)				отмечен
translate interface (переводить интерфейс)				отмечен

5. Перейдите на страницу Administer (Администрирование)→Site configuration (Настройка сайта)→Languages (Языки) (*admin/settings/language*) и щелкните на вкладке Add language (Добавить язык) (*admin/settings/language/add*), чтобы открыть форму, изображенную на рис. 8.11.

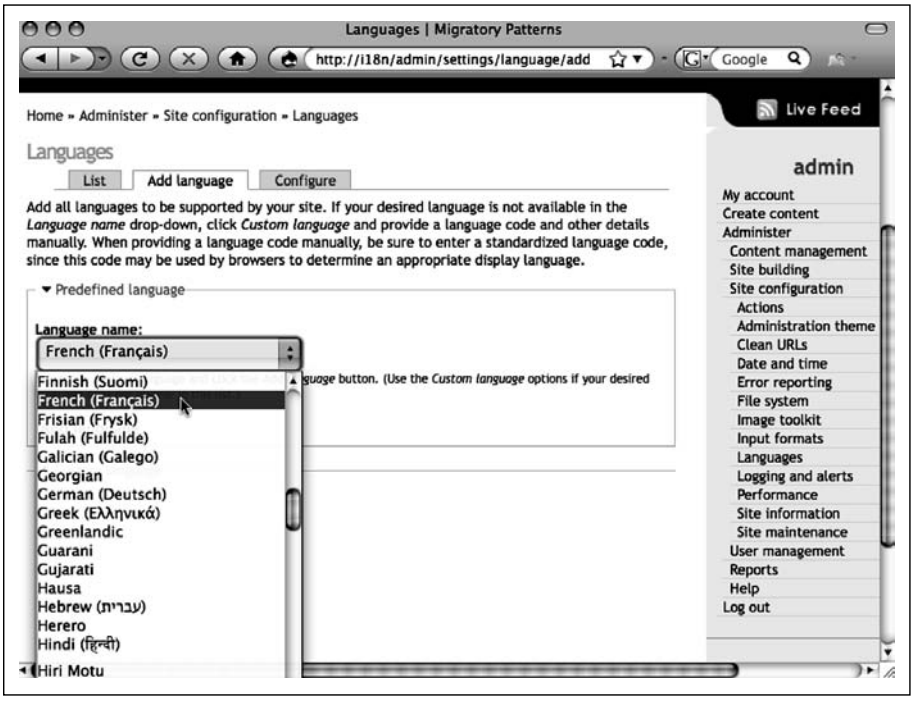


Рис. 8.11. *Форма добавления языка с раскрывающимся списком*

6. Выберите язык из раскрывающегося списка Language name (Название языка) и щелкните на кнопке Add language (Добавить язык).
7. Файлы перевода будут импортированы в Drupal, и вы увидите сообщение Importing interface translations (Импорт переводов интерфейса) вместе с компонентом, отображающим ход выполнения операции.

Если файлы перевода были добавлены после добавления языка на вкладке Add language (Добавить язык), файлы не будут импортированы. В этом случае необходимо удалить язык на вкладке List (Список), на странице Administer (Администрирование)→Site configuration (Настройка сайта)→Languages (Языки) (*admin/settings/language*) и затем добавить его снова, после импортирования файлов перевода.

Повторите эту последовательность действий для каждого языка, который необходимо сделать доступным на сайте. По окончании на странице

Languages (Языки) в списке в виде таблицы будут перечислены все языки, что позволит вам производить такие манипуляции, как отключение языка, выбор языка по умолчанию, изменение порядка следования языков в списке и их удаление. На рис. 8.12 показана таблица на нашем сайте, содержащая три языка: английский, голландский и французский. Английский язык выбран как язык по умолчанию, то есть он будет использоваться как резервный язык в случае, когда язык не определен, и в случае отсутствия перевода для языка, определяемого настройками выбора или настройками на странице My account (Мой профиль). Параметр Weight (Вес) для каждого из языков позволяет определить порядок следования языков, в котором они будут появляться при перечислении, например в раскрывающихся списках. Значение веса здесь имеет тот же смысл, что и в других компонентах Drupal, – элементы с отрицательным весом считаются более «легкими», вследствие чего они будут «всплывать» вверх (или в начало горизонтального списка).

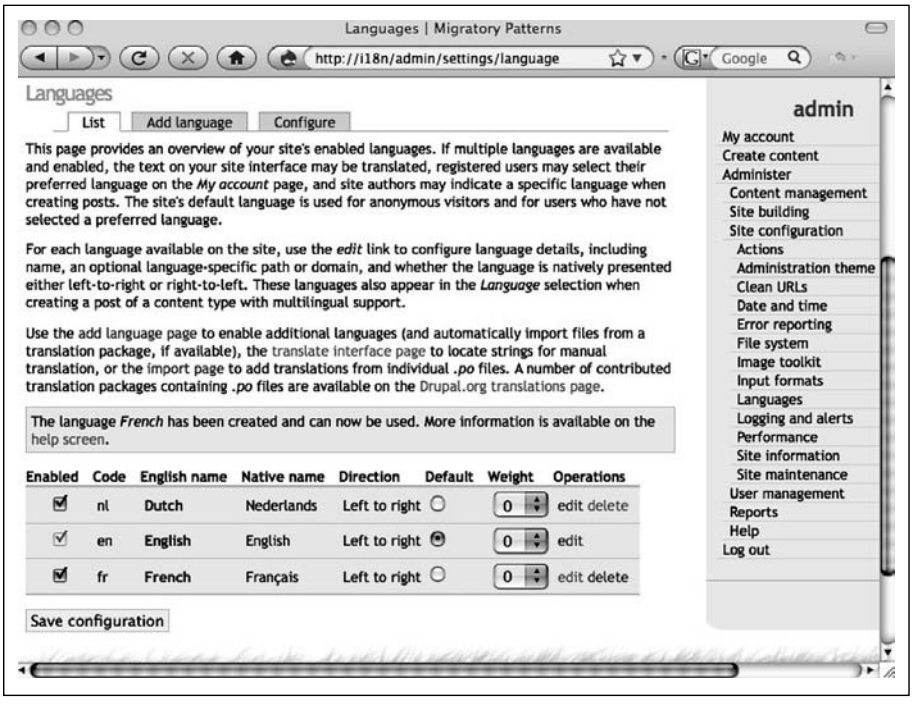


Рис. 8.12. *Список установленных языков*

Имеется также возможность изменять названия языков, их идентификаторы и направление письма, как показано на рис. 8.13, для чего нужно щелкнуть на ссылке edit (изменить). Вообще эти настройки не следует изменять, не имея на то веских причин; и мы тоже оставим все эти настройки со значениями по умолчанию.

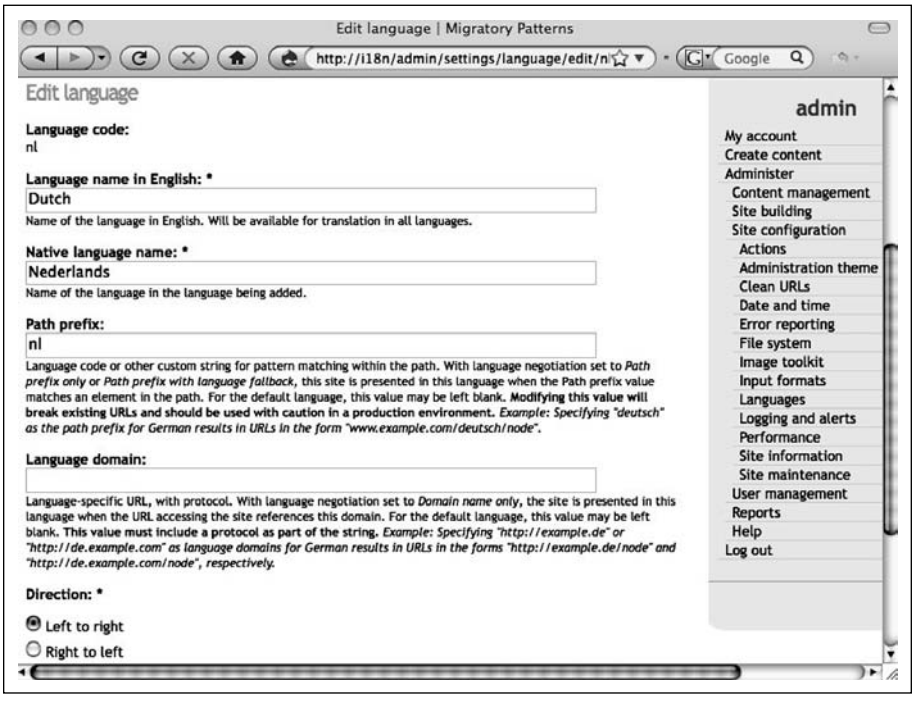


Рис. 8.13. *Страница изменения настроек установленных языков*

Практика: настройка модуля Locale

Теперь нам следует выбрать, как и когда Drupal будет использовать новые языки. Чтобы упростить просмотр содержимого сайта на различных языках и дать пользователям возможность выбирать свой язык, мы добавим на сайт простой переключатель языков.

Настройка механизма выбора языка

Как говорилось выше, в разделе «Переключение языка», у нас имеется несколько способов организовать выбор. Для нас нежелательно вынуждать пользователей вручную выбирать нужный язык через настройки учетной записи (выбрано значение None (Нет)), и у нашего клиента нет отдельных доменных имен для каждого языка (выбрано значение Domain name only (Доменное имя)). В результате у нас остаются два варианта с префиксами пути. Мы будем использовать вариант Path prefix with language fallback (Префикс пути с определением языка) вместо Path prefix only (Префикс пути), потому что наш клиент хочет быть уверенным, что в случае выбора пользователем языка в настройках своей учетной записи он будет получать содержимое на указанном языке, а не на языке по умолчанию, который должен использоваться, только если выбор на основе префикса не дает приемлемого результата:

1. Щелкните на вкладке Configure (Настроить) на странице Languages (Языки) (*admin/settings/language/configure*).
2. Выберите радиокнопку Path prefix with language fallback (Префикс пути с определением языка) и затем щелкните на кнопке Save settings (Сохранить настройки).

Переключатель языка

Модуль Locale предоставляет блок, позволяющий переключать языки, который настраивается точно так же, как любой другой блок, с заданием другого (или отмены) заголовка и различных параметров отображения.

1. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Blocks (Блоки) (admin/build/block) и найдите в списке блок Language switcher (Переключатель языка).
2. Отбуксируйте его мышью в область Right sidebar (Правая колонка) и щелкните на кнопке Save blocks (Сохранить блоки).
3. Теперь вы должны увидеть новый блок с заголовком Languages (Языки), содержащий список всех языков, установленных на сайте, как показано на рис. 8.14.

Рис. 8.14. *Страница сайта с включенным блоком переключения языка*

Практика: переводимое содержимое

Теперь мы можем настроить поддержку нескольких языков для каждого имеющегося типа содержимого (включая любые нестандартные типы). На нашем сайте имеется три типа содержимого: страницы книг, темы форума и заметки. Нам необходимо включить поддержку нескольких языков для типов Story (Статья) и Book page (Страница подшивки), а для тем форумов нам не требуется создавать переводы, поэтому они должны только идентифицировать язык. Нам также необходимо дать всем зарегистрированным пользователям разрешение добавлять переводы.

1. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Modules (Модули) (*admin/build/modules*) и включите следующие модули:
   - Core – optional (Ядро – дополнительные)
     - Book
     - Content translation
2. Затем перейдите на страницу Administer (Администрирование)→Content management (Содержание)→Content types (Типы материалов) (*admin/content/types*) и щелкните на ссылке edit (изменить) для типа Book page (Страница подшивки).

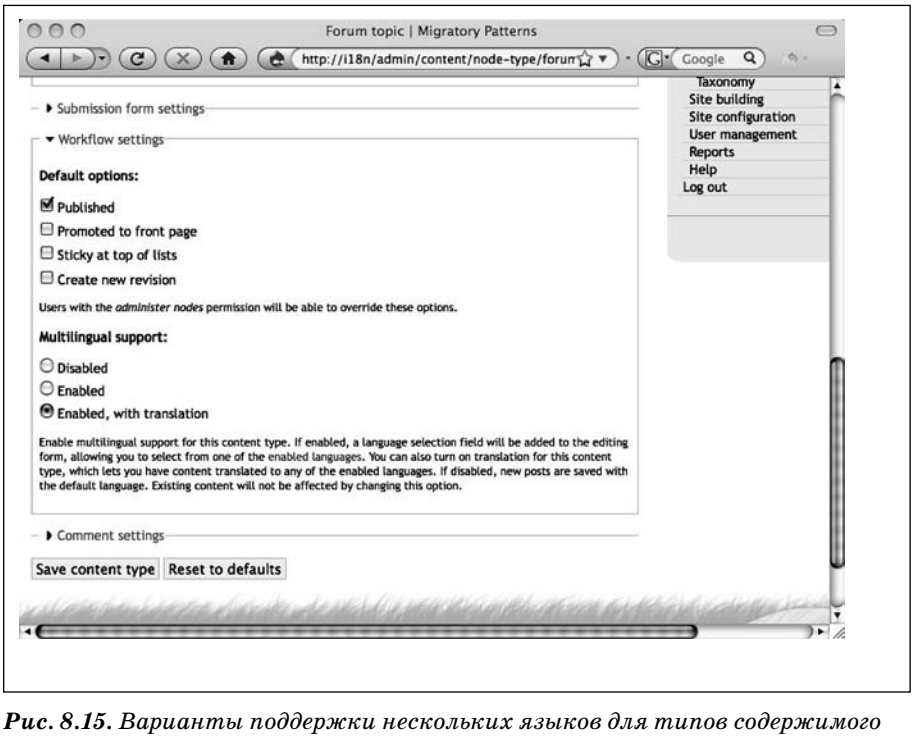


Рис. 8.15. *Варианты поддержки нескольких языков для типов содержимого*

3. Прокрутите страницу вниз и распахните раздел Workflow settings (Параметры). Вы увидите несколько вариантов выбора в параметре Multilingual support (Поддержка многоязычности). Выберите радиокнопку Enabled, with translation (Включено, с переводом), как показано на рис. 8.15, и щелкните на кнопке Save content type (Сохранить тип содержимого). Это позволит идентифицировать язык для каждого элемента содержимого данного типа, а также добавлять переведенные версии, которые будут привязываться к оригиналу автоматически.
4. Повторите те же действия для типа содержимого Story (Статья).
5. Для типа Forum (Форум) выберите вариант Enabled (Включено).
6. Нам необходимо дать всем пользователям возможность переводить содержимое сайта, поэтому следует дать соответствующее разрешение всем пользователям, за исключением анонимных. Так как все пользователи проходят регистрацию, мы можем сделать это, дав разрешение всем зарегистрированным пользователям. Перейдите на страницу Administer (Администрирование)→User management (Управление пользователями)→Permissions (Разрешения) (*admin/user/permissions*), установите разрешения в соответствии с табл. 8.2 и щелкните на кнопке Save permissions (Сохранить права доступа).

Таблица 8.2. Разрешения для модуля Content Translation

Разрешение: модуль translation	**anonymous user (анонимный пользователь)**	**authenticated user (зарегистрированный пользователь)**	**editor (редактор)**	**site administrator (администратор сайта)**
translate content (переводить материалы)		отмечен		

В центре внимания: модуль Localization Client

Вы могли заметить, что даже после установки переводов то здесь то там встречаются непереведенные фрагменты текста. Чаще всего это происходит после установки сторонних модулей. Практически ни один сайт «из коробки» не обеспечивает 100-процентный перевод, поэтому вам наверняка потребуется самостоятельно перевести некоторые элементы. Для этого в Drupal имеется встроенный механизм на основе модуля Locale, но пользоваться им неудобно. Это как раз тот случай, когда во всем блеске проявляет себя сторонний модуль Localization Client. Он делает возможность исправления перевода простой и понятной.

Модуль Localization Client добавляет прямо в нижней части страницы редактор для ввода переведенного текста, который остается с вами при перемещении по сайту. Его можно свернуть, когда он не нужен, и раз-

вернуть, когда потребуется ввести перевод. На рис. 8.16 показана страница с развернутым редактором. Он имеет приятный и простой в использовании интерфейс, позволяющий увидеть, какие строки на странице были переведены и, что более важно, какие не были переведены. Редактор дает возможность просматривать или выполнять поиск по списку – вам остается просто выбрать нужную строку и добавить ее перевод в поле ввода справа.

Следует сделать несколько важных замечаний, которые нужно иметь в виду. Модуль Localization Client – это замечательный инструмент, но он не может работать при выключенной поддержке JavaScript и, к моменту написания этих строк он не позволял переводить строки, добавленные модулем Internationalization (который будет рассматриваться ниже). Кроме того, он может переводить только тот текст, который доступен пользователю, запустившему Localization Client. Например, если текст видим только при определенных условиях, его перевод может не выполняться с помощью Localization Client.

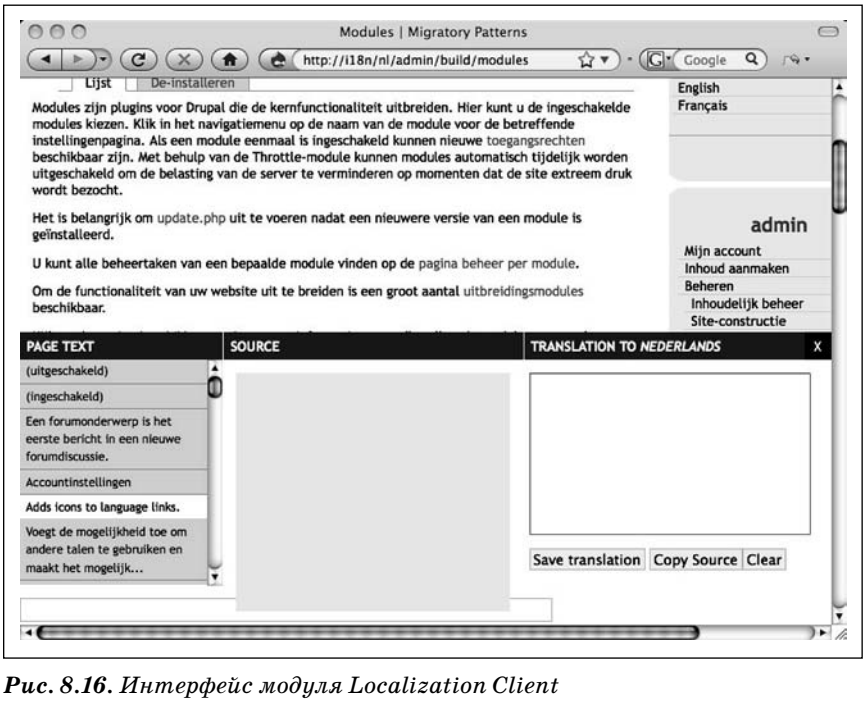


Рис. 8.16. *Интерфейс модуля Localization Client*

Практика: перевод интерфейса

Далеко не всем хочется или требуется устанавливать еще один модуль, а из-за замечаний относительно модуля Localization Client желательно иметь представление о том, как перевод выполняется ядром; об этом мы сейчас и расскажем.

Использование модуля Locale

Перевод интерфейса выполняется с помощью модуля Locale, который должен быть уже включен к моменту установки перевода.

1. Для начала перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Translate interface (Перевод интерфейса) (*admin/build/translate*).
2. Перед вами должна появиться таблица со списком языков и процентами переведенных строк. Щелкните на вкладке Search (Поиск) (*admin/build/translate/search*).
3. Страница поиска строк, изображенная на рис. 8.17, позволяет находить определенные фрагменты текста, находящиеся где-то в интерфейсе Drupal, и переводить их.

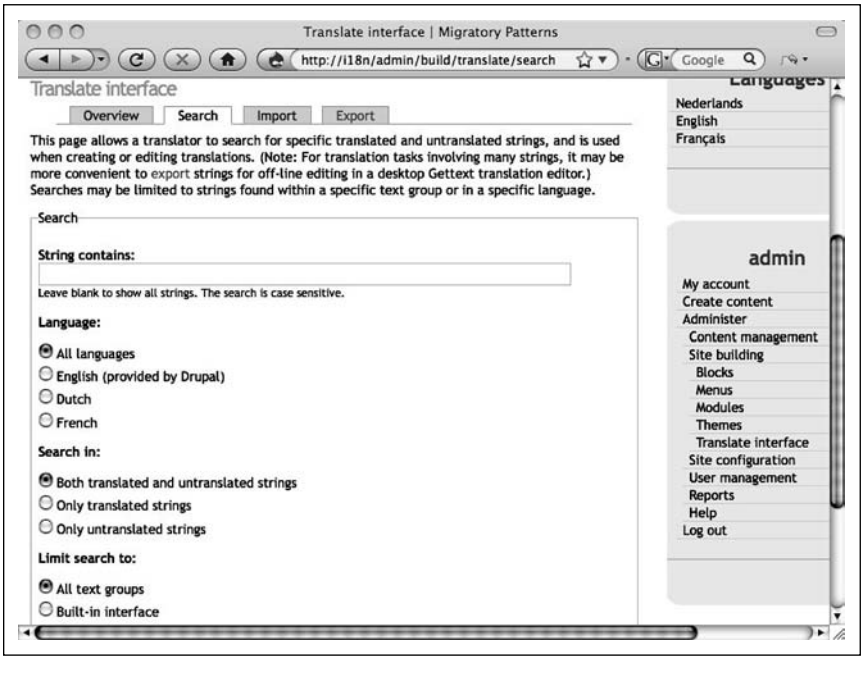


Рис. 8.17. *Страница поиска строк при переводе интерфейса*

4. Введите строку `multilingual support` в поле поиска, оставьте остальные параметры со значениями по умолчанию и щелкните на кнопке Search (Поиск).

Следует сделать два замечания, касающиеся поиска строк для перевода. Во-первых, поиск выполняется с учетом регистра символов. Поиск слова «User» даст иные результаты, нежели поиск по слову «user». Во-вторых, страница с искомой строкой должна

быть открыта хотя бы один раз после включения модуля Locale, в противном случае функция поиска не сможет отыскать требуемый текст.

5. На странице с результатами поиска, изображенной на рис. 8.18, вы увидите список всех мест на сайте, где встречается искомая строка. В колонке Languages (Языки) перечислены коды языков. Если код зачеркнут, это означает, что перевод на данный язык пока отсутствует. В данном примере поиск дал несколько результатов, причем во всех случаях имеются переводы на все поддерживаемые языки.

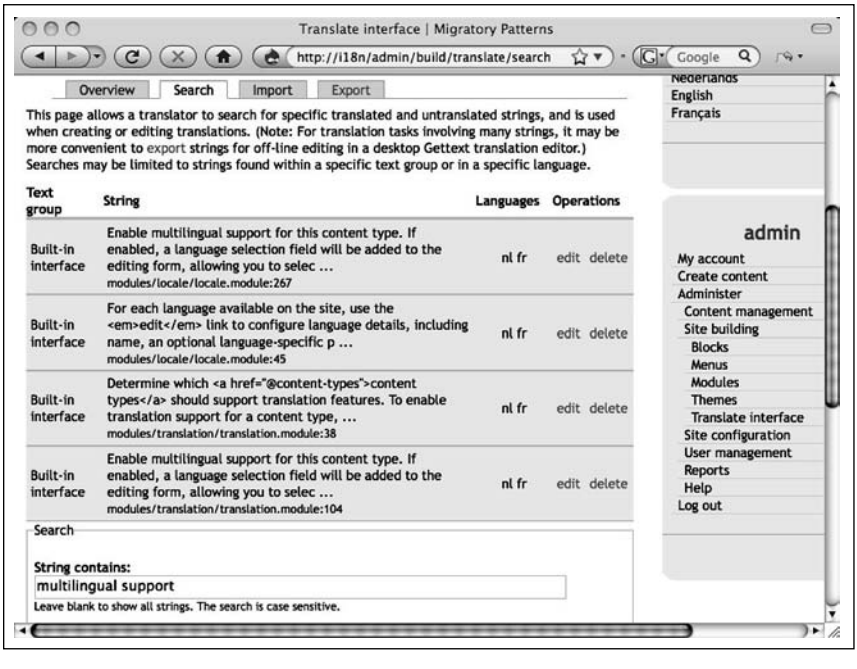


Рис. 8.18. *Результаты поиска строки «multilingual support»*

6. Щелкните на ссылке edit (изменить) для одной из строк, и вы окажетесь на странице перевода, как показано на рис. 8.19. Для каждого языка будет предоставлена отдельная область редактирования, где вы можете добавлять или изменять текст. После щелчка на кнопке Save translations (Сохранить переводы) будет выполнено обновление строк.

Использование модуля Localization Client

Как было показано в предыдущем разделе, сначала нужно увидеть строки, требующие перевода, запомнить их, перейти на страницу перевода, отыскать их и только потом получить возможность перевести их. Получается достаточно сложный и утомительный процесс. К счастью, существует более простой путь: задействовать модуль Localization Client.

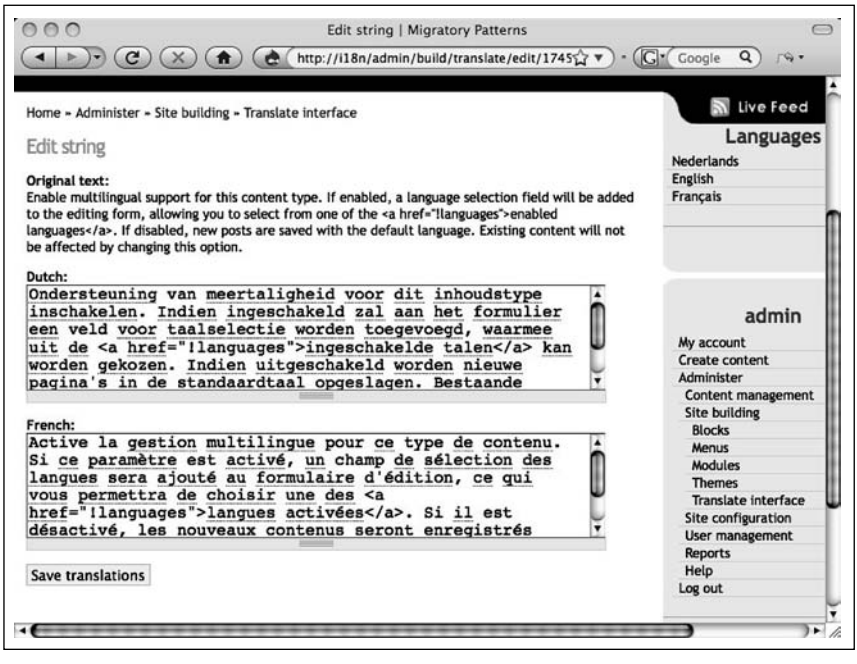


Рис. 8.19. *Страница перевода строки, где отображаются оригинальная строка и перевод для каждого языка*

1. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Modules (Модули) (*admin/build/modules*) и включите модуль:
   - Other (Другой)
     – Localization Client
2. Перейдя на страницу Administer (Администрирование)→User management (Управление пользователями)→Permissions (Разрешения) (*admin/user/permissions*), установите разрешения в соответствии с табл. 8.3 и щелкните на кнопке Save permissions (Сохранить права доступа).

Таблица 8.3. Разрешения для модуля Localization Client

Разрешение: модуль l10n_client module	**anonymous user (анонимный пользователь)**	**authenticated user (зарегистрированный пользователь)**	**editor (редактор)**	**site administrator (администратор сайта)**
use on-page translation (включить перевод на странице)			отмечен	отмечен

3. Теперь переключитесь на язык, отличный от английского, щелкнув на требуемом названии языка в блоке Language (Язык). В нижней части страницы должен появиться голубой прямоугольник с черной кнопкой Translate Text (Перевести текст) справа. На рис. 8.20 она обведена овалом.

Рис. 8.20. *Кнопка Translate Text, созданная модулем Localization Client*

4. Щелкните на кнопке Translate Text (Перевести текст), в результате в нижней части страницы откроется область перевода, как показано на рис. 8.21. Все элементы, перечисленные под заголовком Page Text (Текст страницы), выделенные зеленым цветом, уже имеют перевод. Элементы, выделенные белым цветом, требуют перевода.
5. Имеется возможность сократить список за счет использования поля поиска внизу списка. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Modules (Модули) (*admin/build/modules*), если вы уже покинули ее, и введите слово «localization». По мере ввода список будет изменяться, и в конечном итоге в нем должна остаться строка «Provides on-page localization» (Обеспечивает локализацию страницы), которая является строкой описания, содержащейся в модуле Localization Client. Если кто-то

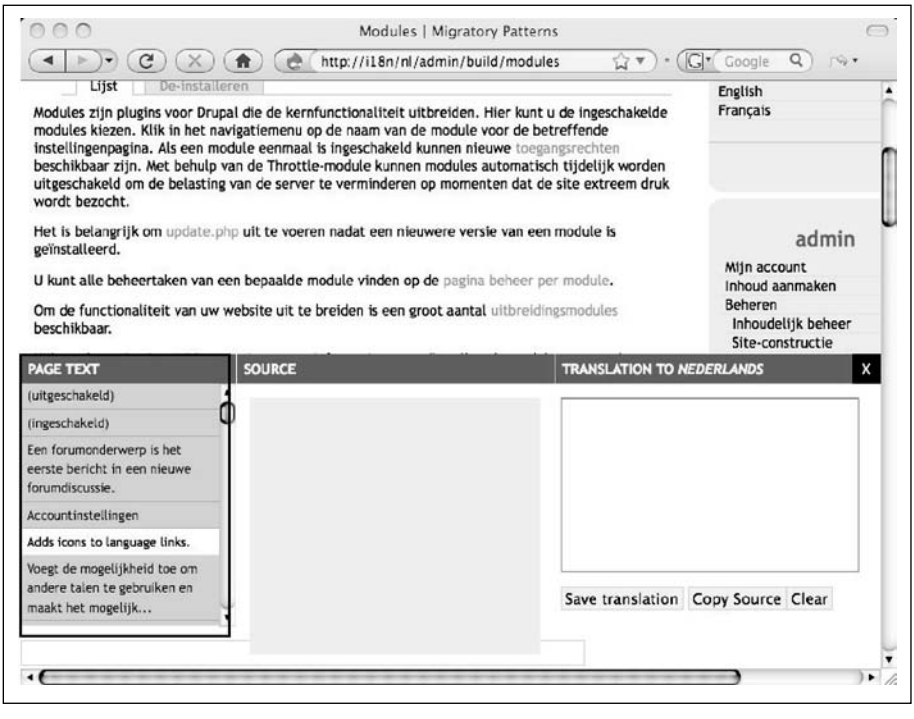


Рис. 8.21. *Интерфейс перевода модуля Localization Client отображает переведенные и непереведенные строки*

еще не предоставил файл перевода для модуля на вашем языке, английский текст в результатах поиска должен отображаться белым цветом (не переведено).

6. После щелчка на элементе списка Page Text (Текст страницы) он будет скопирован в поле Source (Исходный), чтобы вы знали, какой текст переводится. Если элемент уже имеет перевод, он появится в поле Translation to language (Перевод).
7. Вы можете добавить или изменить перевод и затем щелкнуть на кнопке Save translation (Сохранить перевод). Например, «Maakt lokalisatie op de pagina zelf mogelijk» – это перевод данной строки на голландский язык. На рис. 8.22 показан интерфейс модуля Localization Client с выбранной строкой на английском языке и с переводом на голландский язык, готовым к сохранению.
8. По окончании щелкните на изображение крестика в заголовке интерфейса перевода, чтобы закрыть его.
9. Если теперь перезагрузить страницу, вы сможете заметить внесенные изменения.

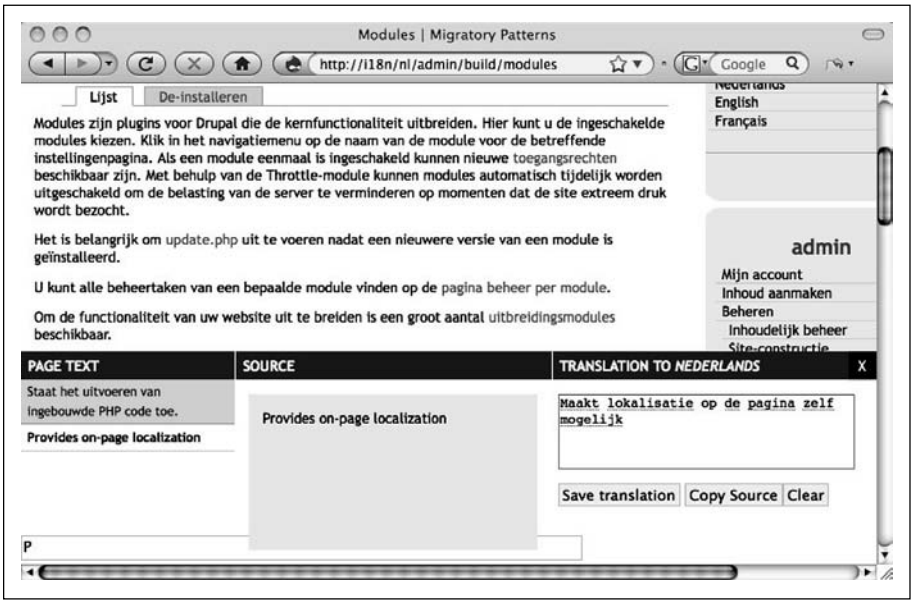


Рис. 8.22. *Использование модуля Localization Client для перевода строки на голландский язык*

Практика: перевод содержимого

Теперь нам необходимо настроить раздел нашего сайта Knowledge Base (База знаний). Мы сначала создадим несколько материалов, а затем покажем, как перевести их.

Перевод

Ранее мы включили поддержку переводов для наших страниц, а теперь пройдем сам процесс перевода:

1. Создайте страницу введения для базы знаний на английском языке. Перейдите на страницу Create content (Создать материал)→Book page (Страница подшивки), заполните поля формы в соответствии с табл. 8.4 и щелкните на кнопке Save (Сохранить).

Таблица 8.4. Содержимое страницы подшивки

Поле	**Значение**
Title (Заголовок)	Introduction (Введение)
Menu settings (Настройки меню)	
Menu link (Заголовок ссылки в меню)	Knowledge Base (База знаний)

Parent item (Родительский пункт)	<Primary Links> (<Основные ссылки>)
Language (Язык)	English (Английский)
Body (Содержимое)	This knowledge base is a place to organize useful information. Anyone with a Migratory Patterns account can add to this book! You may create a new page by clicking the «Add child page» link at the bottom of any existing KB page or by going to Create content→Book page in your navigation block. (Эта база знаний является местом размещения полезной информации. Любой, имеющий учетную запись на сайте Migratory Patterns, сможет добавлять свою информацию в эту подшивку! Вы можете создать новую страницу, щелкнув на ссылке «Добавить дочернюю страницу» внизу существующей страницы БЗ или перейдя на страницу Содержание→Страница подшивки с помощью блока навигации.)
Book outline (Оглавление подшивки)	
Book (Подшивка)	<create a new book> (<создать новую подшивку>)

2. После сохранения содержимого вы увидите вкладку Translate (Переводы) рядом с привычными вкладками View (Просмотреть), Edit (Изменить) и Outline (Оглавление) (для подшивок). Щелкните на вкладке Translate (Переводы), и перед вами появится таблица со всеми языками, поддерживаемыми на вашем сайте, как показано на рис. 8.23.
3. Щелкните на ссылке add translation (добавить перевод) для требуемого языка, в результате чего откроется страница с формой, заполненной значениями, отправленными нами ранее. Теперь вы можете отредактировать текст для полей Title (Заголовок), Menu (Меню) и Body (Содержимое). Обратите внимание, что язык уже выбран. Переведите текст на выбранный язык и не забудьте снова выбрать значение <create a new book> (<создать новую подшивку>) в разделе Book outline (Оглавление подшивки), чтобы эта страница с текстом введения стала начальной страницей подшивки для тех, кто просматривает сайт на этом языке.
4. Щелкнув на кнопке Save (Сохранить), чтобы сохранить новую страницу с переводом, вы увидите внизу страницы ссылку, которая приведет вас на версию этой страницы на другом языке.

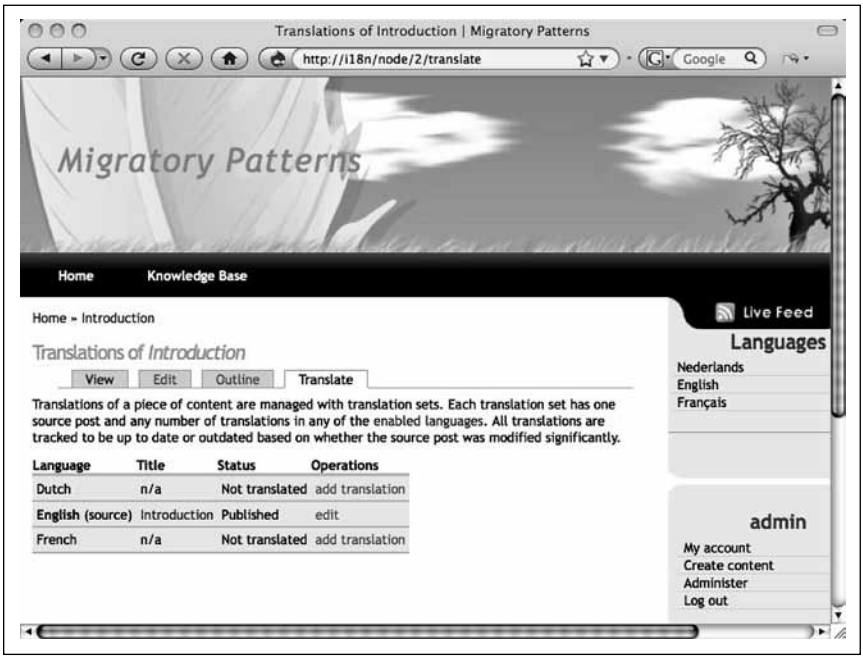


Рис. 8.23. *Вкладка Translate (Переводы) для страницы подшивки, где показаны все доступные языки и переводы*

Сделайте то же самое с элементом содержимого Welcome (Добро пожаловать) на главной странице (если вы использовали профиль Multilingual) или с любой другой страницей, созданной перед включением возможности перевода содержимого. Чтобы добавить версии для других языков, достаточно просто отредактировать существующий материал, выбрав язык и перейдя на вкладку Translate (Переводы), как это делалось выше.

Имейте в виду, что по умолчанию для любого содержимого в качестве настройки языка используется значение Language neutral (Нейтральный по отношению к языку). Чтобы увидеть вкладку Translate (Переводы) и получить возможность добавить перевод, необходимо идентифицировать язык содержимого. Если при создании содержимого язык не был выбран или если модуль Content Translation был включен уже после создания содержимого, для этого содержимого будет выбран язык Language neutral (Нейтральный по отношению к языку).

При создании переводов для любых материалов, публикуемых на главной странице сайта (таких, как сообщение Welcome (Добро пожаловать) в профиле), вы могли заметить, что появляются оба варианта перевода. То же самое происходит с элементами меню нашей базы знаний, как показано на рис. 8.24. Мы решим эту проблему в следующем разделе.

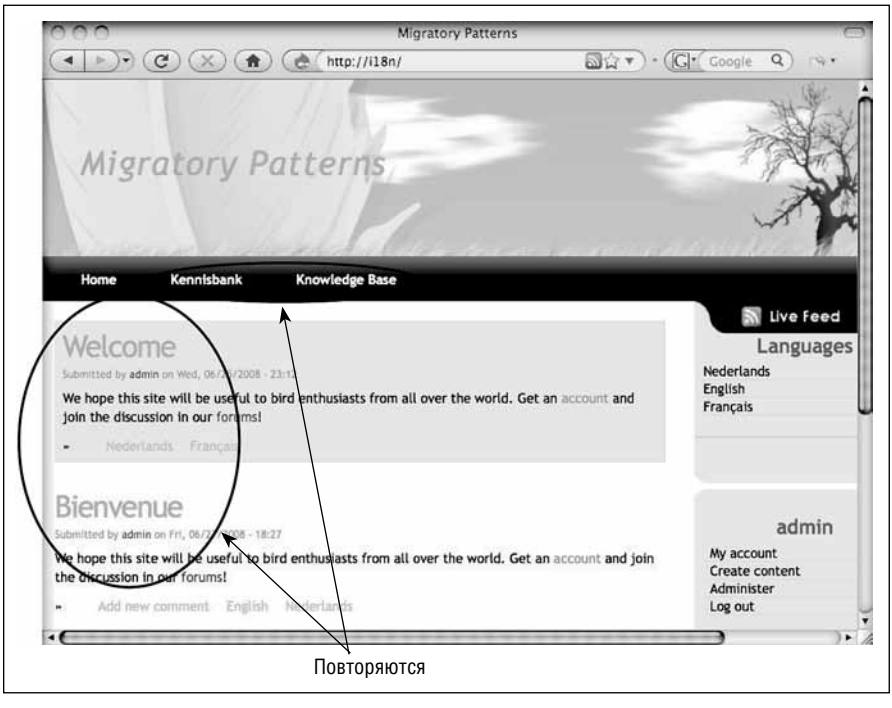


Рис. 8.24. *Повторяющиеся элементы – по одному для каждого языка*

В центре внимания: Internationalization

К настоящему моменту мы заложили неплохую основу для перевода содержимого сайта, но все-таки у нас не все так гладко, как хотелось бы. У нас имеются переводы для некоторых материалов и элементов меню, но все они отображаются одновременно. На главной странице отображаются сообщения и пункты меню на всех языках независимо от того, на каком языке выполняется просмотр сайта. Вы будете также замечать, что некоторый текст по-прежнему отображается на английском языке. Например, щелкнув на пункте меню Create content (Создать материал), можно увидеть, что описание типа содержимого имеет перевод, а имя типа – нет. Чтобы продвинуться в поддержке многоязычности дальше и добавить нашему сайту блеска, мы задействуем пакет модулей Internationalization (i18n). Основу пакета составляет модуль Internationalization, распространяемый с несколькими другими модулями, которые все вместе предназначены для расширения поддержки многоязычности в ядре.

Эти модули позволят получить в интерфейсе создания перевода ядра те строки, которые иначе недоступны. Инструменты ядра Drupal для работы с интерфейсом в состоянии обнаруживать только те строки, кото-

рые жестко зашиты непосредственно в ядро (с использованием функции `t()`), то есть сгенерированные программным кодом. Все строки, созданные пользователем, им недоступны. Важно понять, что строки, добавленные модулями из пакета Internationalization, *не* будут доступны в интерфейсе Translate Text (Перевести текст) модуля Localization Client. Чтобы перевести их, необходимо использовать встроенный интерфейс ядра, описанный выше, в разделе «Практика: перевод интерфейса», находящийся на странице Administer (Администрирование)→Site building (Конструкция сайта)→Translate interface (Переводить интерфейс) (*admin/build/translate*).

Вторая особенность многих из этих модулей заключается в том, что они добавляют возможность выбирать язык для элементов, таких как пункты меню, блоки или рубрики таксономии, как показано на рис. 8.25. Наличие возможности идентифицировать язык для каждого отдельного элемента позволяет модулю Internationalization фильтровать отображаемые элементы по выбранному языку, что приводит к уменьшению беспорядка и снижению числа повторяющихся элементов.

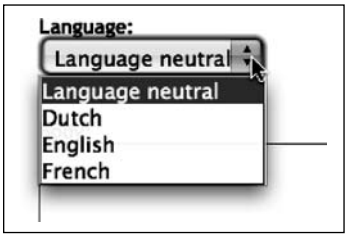


Рис. 8.25. *Компонент выбора языка*

Пакет модулей Internationalization предоставляет множество инструментов, но не все из них нам потребуются при создании сайта Migratory Patterns. Мы расскажем, что из себя представляют эти инструменты, и затем по мере конструирования нашего сайта увидим некоторые из них в действии.

Выбор содержимого

Самое первое, что вы наверняка заметили после добавления первых переводов, это то, что все переведенные элементы отображаются на главной странице независимо от выбранного языка. Модуль Internationalization помогает нам взять это под свой непосредственный контроль, давая возможность определять, как должно отображаться содержимое, имеющее переводы на несколько языков. Имеется три варианта идентификации языка содержимого:

Язык не определен

Принадлежность элемента содержимого к языку явно не указывается. Это может происходить при выборе значения Language neutral

(Нейтральный по отношению к языку) или All languages (Все языки) в списках выбора языка для различных типов содержимого.

Текущий язык

Текущий язык определяется либо автоматически, посредством настройки механизма выбора языка (например, по коду языка, такому как «en», включенному в путь адреса URL), или вручную, когда пользователь выбирает язык отображения с помощью переключателя.

Язык по умолчанию

Это основной параметр настройки веб-сайта, находящийся на странице управления языками Administer (Администрирование)→Site configuration (Настройка сайта)→Languages (Языки) (*admin/settings/language*). Существует только один язык по умолчанию и это тот самый язык, который используется при отключенном механизме автоматического определения языка.

Имеется пять различных режимов выбора, которые можно использовать для настройки критерия выбора отображаемого содержимого. На рис. 8.26 показаны настройки режима, которые будут упоминаться в примерах, следующих ниже.

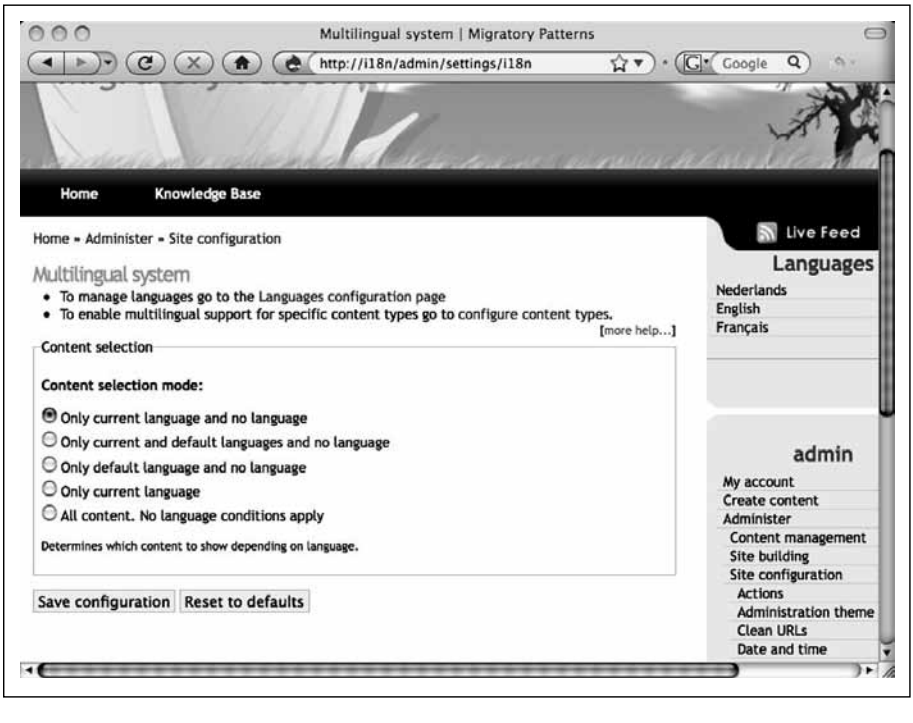


Рис. 8.26. *Настройки режима выбора содержимого в модуле Internationalization*

Режим выбора содержимого определяет порядок фильтрации материалов для отображения на основе языка. Проиллюстрируем на примере поведение различных настроек. На нашем сайте используется три языка: английский, голландский и французский. Языком по умолчанию является английский. Для простоты предположим, что на главной странице у нас имеется «три» элемента содержимого в следующих четырех узлах: Drupal, Welcome, Welkom и Les mots. Элемент Drupal нейтрален по отношению к языку, второй элемент доступен на английском (Welcome) и голландском (Welkom) языках и элемент Les mots – только на французском.

При выборе французского языка в блоке переключателя на главной странице будут отображаться следующие узлы с учетом указанного режима выбора:

- Only current language and no language (Только для текущего языка и когда язык не определен): Drupal (не определен) и Les mots (французский)
- Only current and default languages and no language (Только для текущего языка, для языка по умолчанию и когда язык не определен): Drupal (не определен), Welcome (английский) и Les mots (французский)
- Only default language and no language (Только для языка по умолчанию и когда язык не определен): Drupal (не определен) и Welcome (английский)
- Only current language (Только для текущего языка): Les mots (французский)
- All content (Все материалы): Критерий выбора не используется: отображаются все четыре узла

Если в блоке переключателя выбрать голландский язык:

- Only current language and no language (Только для текущего языка и когда язык не определен): Drupal (не определен) и Welkom (голландский)
- Only current and default languages and no language (Только для текущего языка, для языка по умолчанию и когда язык не определен): Drupal (не определен), Welcome (английский) и Welkom (голландский)
- Only default language and no language (Только для языка по умолчанию и когда язык не определен): Drupal (не определен) и Welcome (английский)
- Only current language (Только для текущего языка): Welkom (голландский)
- All content (Все материалы): Критерий выбора не используется: отображаются все четыре узла

Для простоты в этом примере использовались узлы, но эти настройки применяются ко всем элементам сайта, включая элементы меню, блоки и тому подобное.

Модуль Strings

Модуль Strings необходим для работы большинству модулей в пакете Internationalization. Он не имеет самостоятельного значения, но при совместном использовании с другими модулями пакета Internationalization превращает различные фрагменты текста на сайте в строки, допускающие возможность перевода, которые становятся доступны интерфейсу перевода ядра.

Переменные сайта, зависящие от языка

На сайтах, работающих под управлением Drupal, может иметься множество фрагментов текста, не связанных с какими-либо определенными узлами. Эти фрагменты хранятся в базе данных и называются *переменными*. Примерами таких переменных являются название сайта и слоган, которые можно найти на странице Site Information (О сайте), или шаблоны электронных писем на странице User settings (Параметры регистрации). Нет никакого простого способа получить доступ к этим фрагментам текста с помощью обычного интерфейса перевода строк.

Модуль Internationalization добавляет возможность сообщить системе Drupal, что у вас имеется перевод для этих переменных. К сожалению, для обеспечения этой возможности необходимо редактировать файл *settings.php*. Однако после добавления необходимой информации в настройки можно продолжать использовать обычный интерфейс перевода. Более подробно мы поговорим об этом в следующем разделе «Практика: особенности пакета Internationalization».

Вспомогательные модули

Модуль Internationalization взаимодействует с некоторыми встроенными модулями, обеспечивая помощь в их переводе:

Menu

Модуль Multilingual Menu позволяет добавлять любые нестандартные пункты, создаваемые вами, в список строк в интерфейсе перевода. С его помощью можно также указать принадлежность пункта меню к языку, в результате чего необходимость отображения такого пункта будет определяться правилом, выбранным для отображения содержимого. Обратите внимание, что независимо от того, используется этот модуль или нет, имеется возможность создать пункт меню для узла на каждом языке, который будет подчиняться правилам отображения. То есть, если вам требуется всего лишь создавать пункты меню для узлов, этот модуль можно не использовать. Если же вам требуется создать пункты меню, не указывающие на определенные узлы, то этот модуль позволит вам создавать переведенные версии.

Taxonomy

Модуль Multilingual taxonomy предоставляет несколько вариантов слежения за переводами рубрик таксономии. При создании нового словаря имеется возможность указать, будут ли переводиться названия рубрик с помощью обычного интерфейса перевода, или для каждого языка будут определяться независимые наборы рубрик, или весь словарь будет принадлежать одному языку. Если выбирается вариант создания отдельных наборов рубрик для каждого языка, появляется возможность выбирать язык для каждой рубрики. После создания рубрик и назначения языка для каждой из них можно определить ассоциации между ними. Например, рубрики «cat» (для которой выбран английский язык) и «le chat» (для которой выбран французский язык) могут быть помечены как эквивалентные.

Block

Модуль Multilingual Blocks позволяет выбирать язык для каждого блока. Назначение языка для блока будет определять порядок его отображения в соответствии с настройками отображения основного содержимого. Для нестандартных блоков, создаваемых вами, можно также указать, будет ли содержимое блока переводиться посредством добавления строк в интерфейс перевода.

Profile

Встроенный модуль Profile позволяет добавлять поля, которые будут заполняться пользователями в своих учетных данных. Модуль Multilingual Profile гарантирует, что атрибуты добавляемых полей, такие как название, описание и тому подобное, будут доступны для перевода. Примечательно, что этот модуль не обеспечивает доступность для перевода содержимого, вводимого пользователями в эти поля.

Poll

Каждый перевод голосования – это новый узел, а механизм голосования в системе Drupal изначально отслеживает число голосов отдельно для каждого узла. Модуль Multilingual Poll гарантирует, что результаты голосования для всех переводов одного и того же голосования будут объединяться, чтобы получить точные результаты голосования независимо от того, на каком языке это голосование просматривается.

Синхронизация

Последний модуль в пакете Internationalization – это модуль Translation Synchronization. Данный модуль обеспечивает синхронизацию таксономии и полей узлов между несколькими переводами. Например, если у вас имеется некоторый элемент содержимого, такой как сообщение в блоге, на трех языках и для него определена рубрика, то при изме-

нении рубрики в одном узле этот модуль обеспечит изменение рубрики в двух других узлах.

Если для словаря выбрана настройка языка None (Нет) или Localize terms (Переводить термины), то рубрики будут просто копироваться из оригинального узла. Если выбрана настройка Per language terms (Термины для каждого языка), то рубрики будут изменяться в соответствии с переводами.

Практика: особенности пакета Internationalization

Выбор содержимого

По умолчанию всегда отображается переведенное содержимое. Основной модуль Internationalization позволит вам обеспечить отображение только тех материалов, которые соответствуют текущему языку:

1. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Modules (Модули) (*admin/build/modules*) и включите следующий модуль:
   - Multilanguage – i18n (Многоязычная поддержка – i18n)
     – Internationalization
2. Перейдите на страницу Administer (Администрирование)→User management (Управление пользователями)→Roles (Роли) (*admin/user/roles*).
3. Затем щелкните на ссылке edit permissions (изменить права) для роли site administrator, для модуля i18n отметьте разрешение administer all languages (управлять всеми языками) и сохраните форму.
4. Перейдите на страницу Administer (Администрирование)→Site configuration (Настройка сайта)→Multilingual system (Многоязычная поддержка) (*admin/settings/i18n*) и найдите группу параметров Content selection (Выбор материала). По умолчанию в этой группе должен быть выбран параметр Only current language and no language (Только для текущего языка и когда язык не определен). Убедитесь, что выбран этот параметр и сохраните настройки.

Если теперь перейти на главную страницу, можно увидеть, что на ней присутствует единственный элемент Welcome (Добро пожаловать) и единственный пункт меню Knowledge Base (База знаний). Если в блоке переключения языка выбрать пункт Nederlands (Голландский), на главной странице появится версия элемента Welkom на голландском языке, а элементы меню будут замещены версиями на голландском языке.

Теперь на главной странице отсутствуют повторяющиеся элементы, что показано на рис. 8.27. Далее перейдем к другим элементам сайта, сложнее поддающимся переводу, таким как название сайта, Migratory Patterns.

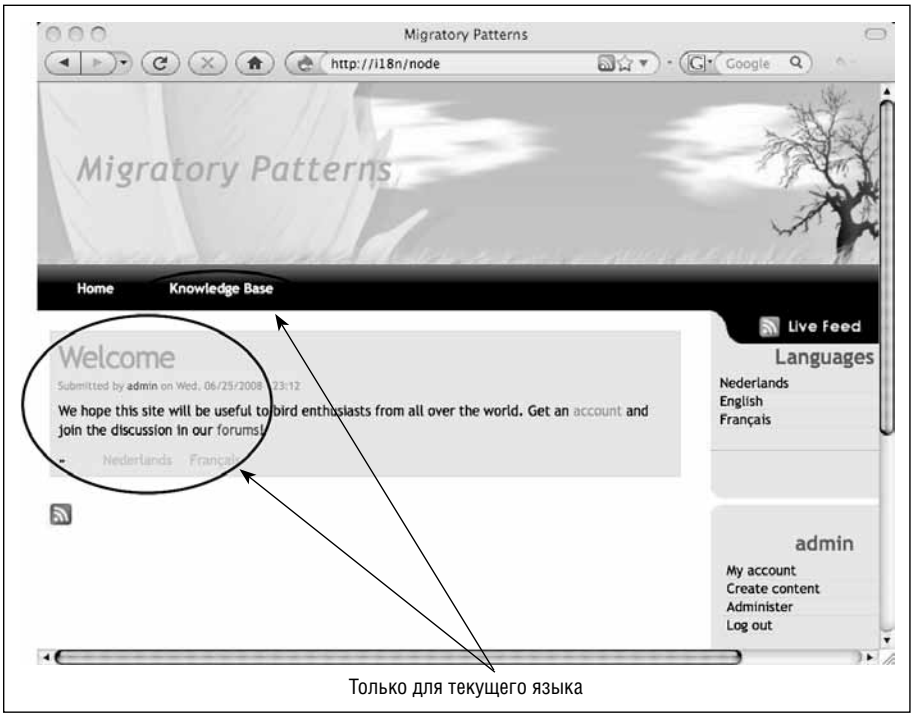


Рис. 8.27. *При включенном режиме выбора содержимого мы больше не видим повторяющиеся элементы*

Переменные сайта

Прежде чем переводить различные переменные сайта, необходимо сообщить системе Drupal, какие из них необходимо сделать доступными для перевода. Для этого нужно отредактировать файл *settings.php*:

1. Перейдите в подкаталог *sites/default* в каталоге установки Drupal.
2. В настоящий момент файл *settings.php* доступен только для чтения, поэтому сначала нужно изменить права доступа к файлу, сделав его доступным для записи. (В разных операционных системах это выполняется по-разному. Например, в Linux нужно установить разрешения файла в значение `666` или `a+w`, а в Windows – снять флажок `Read-only` (Только чтение) в свойствах файла.)
3. Откройте файл *settings.php* в текстовом редакторе и переместитесь в конец файла. Мы добавим в файл новый параметр настройки для модуля Internationalization и определим для примера единственную переменную, хранящую название сайта. Эту информацию и список переменных можно найти в файле *INSTALL.txt* модуля Internationalization. Добавьте следующие строки в конец файла *settings.php*:

```
$conf['i18n_variables'] = array(
  'site_name',
);
```

4. Конец файла теперь должен выглядеть, как показано на рис. 8.28.
5. Сохраните файл и измените права доступа к файлу, сделав его доступным только для чтения. (И снова в разных операционных системах это делается по-разному. В Linux можно опять установить разрешения файла в значение 444 или a-w, а в Windows – отметить флажок Read-only (Только чтение) в свойствах файла.)
6. Теперь, чтобы выполнить перевод, перейдите на страницу Administer (Администрирование)→Site configuration (Настройка сайта)→Site information (О сайте) (*admin/settings/site-information*), где вы увидите, что справочный текст под полем Name (Имя) теперь сообщает: «This is a multilingual variable» (Значение этой переменной может быть определено на нескольких языках), как показано на рис. 8.29.
7. Выберите другой язык в блоке переключения языка.
8. Измените название сайта в поле Name (Имя) и щелкните на кнопке Save configuration (Сохранить конфигурацию).

Теперь можно увидеть, что название сайта изменилось в соответствии с введенным именем. Если затем переключиться на английский или какой-нибудь другой язык, мы увидим, что название сайта опять стало прежним.

Простое переопределение строк

При добавлении переменных для перевода вы, возможно, обратили внимание на закомментированный раздел «String overrides» (Переопределение строк) в файле *settings.php*. Вы можете использовать эти параметры настройки для переопределения на сайте любых, жестко зашитых строк на английском языке. Этот набор параметров обычно не принято использовать для создания полностью локализованных или многоязычных сайтов, потому что для этого в файле *settings.php* необходимо будет перечислить все строки по отдельности. Но он отлично подходит для случаев, когда требуется быстро изменить небольшое количество строк и при этом не хочется возиться с установкой «еще одного модуля» или проделывать стандартную процедуру перевода строк.

Отличный пример использования этих параметров настройки приводится прямо в файле *settings.php*: изменение слова «forum» (форум) на другое, более подходящее для пользователей, такое как «discussion board» (доска обсуждения).

Для изменения этих значений из интерфейса Drupal можно также использовать модуль String Overrides.

```
/**
 * String overrides:
 *
 * To override specific strings on your site with or without enabling locale
 * module, add an entry to this list. This functionality allows you to change
 * a small number of your site's default English language interface strings.
 *
 * Remove the leading hash signs to enable.
 */
# $conf['locale_custom_strings_en'] = array(
#   'forum'      => 'Discussion board',
#   '@count min' => '@count minutes',
# );
$conf['i18n_variables'] = array(
  'site_name',
);
```

Рис. 8.28. *Переменная с названием сайта добавлена в файл settings.php*

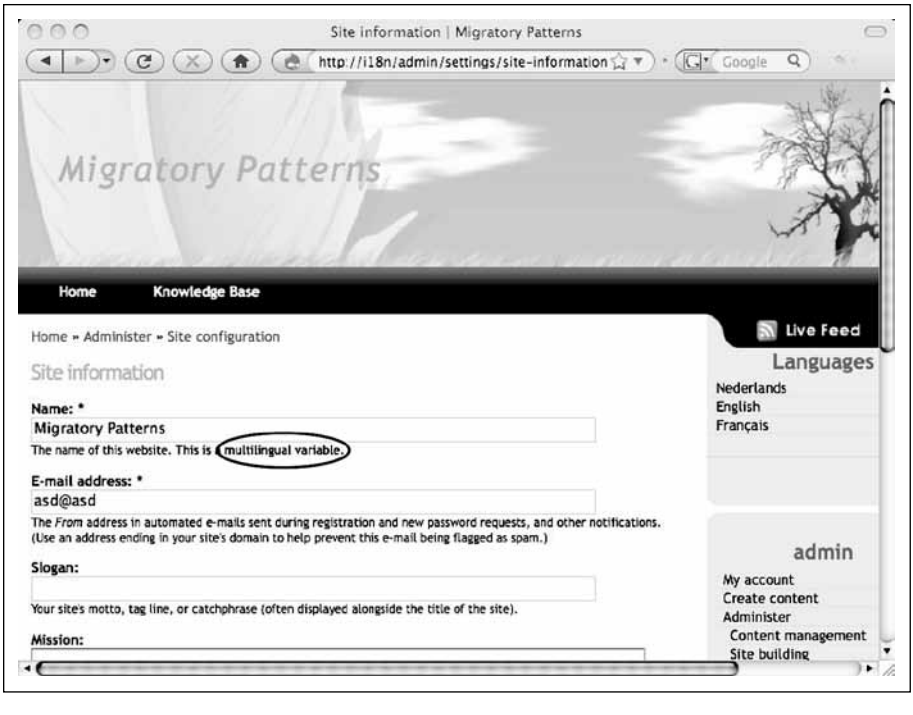


Рис. 8.29. *Теперь переменная site name (название сайта) сообщает, что она может быть определена на нескольких языках*

Какие переменные доступны?

Первый вопрос, который приходит на ум, когда дело доходит до переменных сайта: как узнать, какие переменные доступны? В файле *INSTALL.txt* модуля Internationalization и в руководстве по модулям, которое можно найти на сайте *Drupal.org* (*http://drupal.org/node/134002*), приводится список, пусть и не всех, но наиболее часто используемых переменных. Один из способов узнать, какие переменные доступны, заключается в том, чтобы установить модуль Devel, который предоставляет блок Development (помимо массы других полезных инструментов разработчика). В этом блоке имеется ссылка Variable editor (Редактор переменных), которая ведет к списку всех определенных в настоящий момент переменных.

Имейте в виду, что здесь перечисляются не все возможные переменные, а только те, значения которых были определены. Многие переменные ядра не записываются в базу данных, пока вы не посетите соответствующие страницы и не сохраните соответствующие настройки, поэтому, чтобы увидеть некоторую переменную, значение которой требуется перевести, сначала необходимо перейти на соответствующую страницу и сохранить настройки. Например, чтобы увидеть переменные с текстами сообщений, отправляемых пользователям по электронной почте, сначала нужно перейти на страницу Administer (Администрирование)→User management (Управление пользователями)→User settings (Параметры регистрации) (*admin/user/settings*) и щелкнуть на кнопке Save configuration (Сохранить конфигурацию). Теперь, вернувшись по ссылке Variable editor (Редактор переменных) в блоке Development, вы увидите все вновь добавленные переменные с текстами сообщений, отправляемых пользователям по электронной почте.

Типы содержимого

На странице Create content (Создать материал) при создании новых материалов для наших типов содержимого и для полей по-прежнему используются английские названия.

1. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Modules (Модули) (*admin/build/modules*) и включите модули Strings и Content Types. После включения модулей вы должны увидеть сообщение, указывающее количество вновь добавленных строк.
2. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Translate interface (Переводить интерфейс) (*admin/build/translate*), где вы увидите новый столбец Content type (Тип

материала) в таблице Language (Язык). В этом столбце отображается количество новых строк, созданных модулем Internationalization. Вы можете увидеть новый столбец на рис. 8.30 и сравнить эту таблицу с таблицей на рис. 8.3 выше в этой главе. Процент переведенных строк должен быть равен 0, так как мы не перевели еще ни один из типов содержимого.

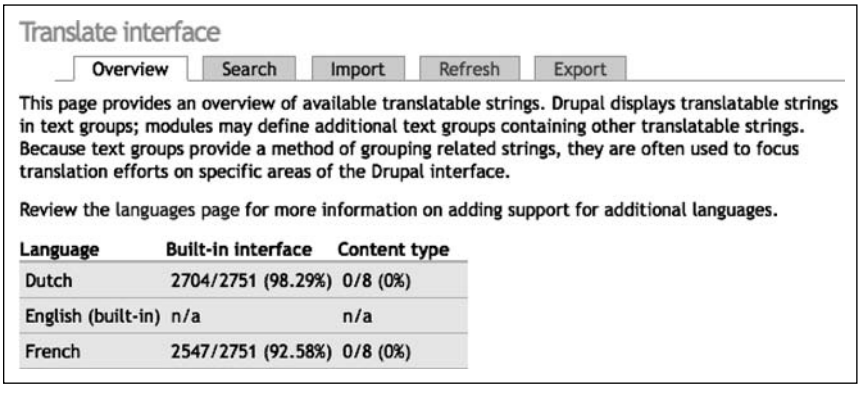


Рис. 8.30. *Страница редактирования строк для типа содержимого «Book page» показывает наличие переводов на два языка*

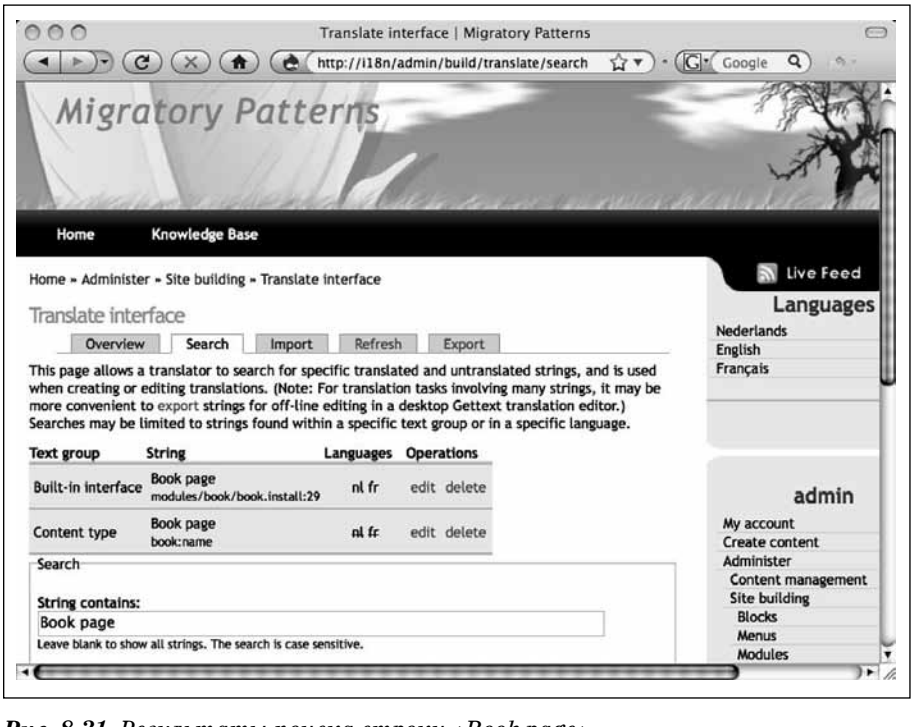


Рис. 8.31. *Результаты поиска строки «Book page»*

3. Щелкните на вкладке Search (Поиск) и отыщите строку «Book page». Вы должны получить два результата, как показано на рис. 8.31: одна из найденных строк относится к группе Built-in interface (Встроенный интерфейс) и уже имеет перевод, а другая, в группе Content type (Тип материала), не имеет перевода.
4. Щелкните на ссылке edit (изменить) для строки Content type (Тип содержимого) и введите перевод Boekpagina. (при желании можно также добавить перевод на французский язык Page de livre)[1], как показано на рис. 8.32.

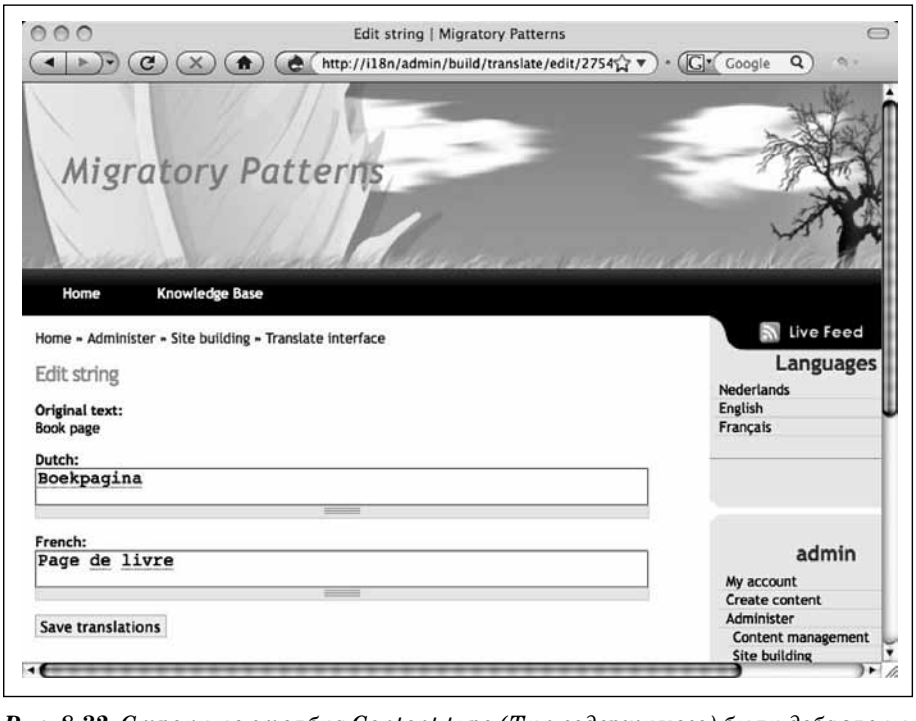


Рис. 8.32. *Строки из столбца Content type (Тип содержимого) были добавлены в таблицу Translate interface (Переводить интерфейс)*

5. Если после сохранения нового перевода перейти на страницу Create content (Создать материал) и включить голландский язык, мы увидим, что теперь тип содержимого «Book page» отображается на голландском языке. Щелкните на ссылке Boekpagina (Страница подшивки), как если бы вы собирались создать новую страницу подшивки. Вы увидите, что название поля «Title» по-прежнему выводится на английском языке. Вы должны также заметить сообщение, сообщающее о создании дополнительных строк.

[1] Или на русский язык: «Страница подшивки» – *Примеч. перев.*

6. Вернитесь на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Translate interface (Переводить интерфейс) (*admin/build/translate*) и измените название поля «Title».
7. Перейдите на вкладку Search (Поиск) и введите строку `Title`. Нам уже известно, что искать изменения необходимо только в группе Content type (Тип содержимого), поэтому выберите радиокнопку Content type (Тип содержимого) в разделе Limit search to: (Ограничить поиск), как показано на рис. 8.33. Затем щелкните на кнопке Search (Поиск).
8. На этот раз мы получили единственный результат. Щелкните на ссылке edit (изменить) в строке с результатом, введите перевод на голландский язык `Titel` (`Titre` – на французском языке)[1] и сохраните его.
9. Вернитесь на страницу Create content (Создать материал) и переключитесь на голландский язык.
10. Щелкните на ссылке Boekpagina, чтобы создать узел типа «Book page», и вы увидите, что поле «Title» теперь называется «Titel».

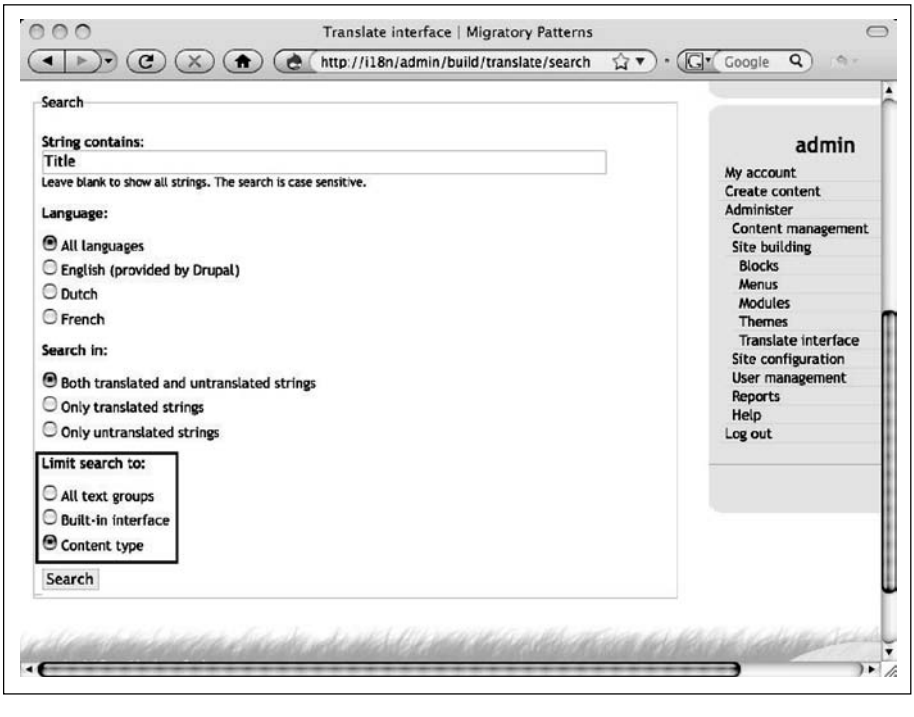


Рис. 8.33. *Ограничение области поиска только типами содержимого*

[1] Или на русский язык: «Заголовок». – *Примеч. перев.*

Таксономия

Форумы в Drupal основаны на использовании таксономии, поэтому нам необходимо найти способ синхронизировать создаваемые нами рубрики, чтобы ограничить действие таксономии только рубриками для данного языка. У нас на сайте уже имеется словарь Forums, поэтому мы сейчас настроим переводы для него.

Форумы

Форумы на сайте будут отображаться с разбивкой на темы, следуя тем же правилам выбора содержимого, которые применяются ко всему остальному содержимому. На сайте сначала будут созданы контейнеры форумов и темы, после чего пользователи смогут обращаться к ним, используя язык по своему выбору.

Использовав профиль установки, мы сейчас имеем контейнер «Types of Birds» (Типы птиц) с двумя форумами в нем: Raptors (Хищные птицы) и Waterbirds (Водоплавающие птицы). Давайте включим поддержку нескольких языков для форумов:

1. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Modules (Модули) (*admin/build/modules*) и включите модуль Multilingual taxonomy.
2. Перейдите на страницу Administer (Администрирование)→Content management (Содержание)→Taxonomy (Таксономия) (*admin/content/taxonomy*) и щелкните на ссылке edit vocabulary (изменить словарь) словаря Forums. На странице с настройками, как показано на рис. 8.34, вы увидите, что появилась новая группа параметров Multilingual options (Настройки многоязычности).
3. Выберите радиокнопку Localize terms (Переводить термины) и сохраните изменения.
4. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Translate Interface (Переводить интерфейс) (*admin/build/translate*), и вы увидите, что в таблице появилась новая колонка Taxonomy (Таксономия). Обратите внимание, что у нас имеется три рубрики (со своими описаниями) для форумов (из профиля установки), а в таблице отмечено, что только одна строка доступна для перевода. Это неправильно. Прежде чем мы сможем перевести все наши строки, мы должны получить их.
5. Вспомните, что прежде чем строки, которые нам требуется перевести, смогут обнаруживаться функцией поиска, необходимо посетить страницу, содержащую их. Вернитесь обратно на страницу управления таксономией Administer (Администрирование)→Content management (Содержание)→Taxonomy (Таксономия) (*admin/content/taxonomy*).

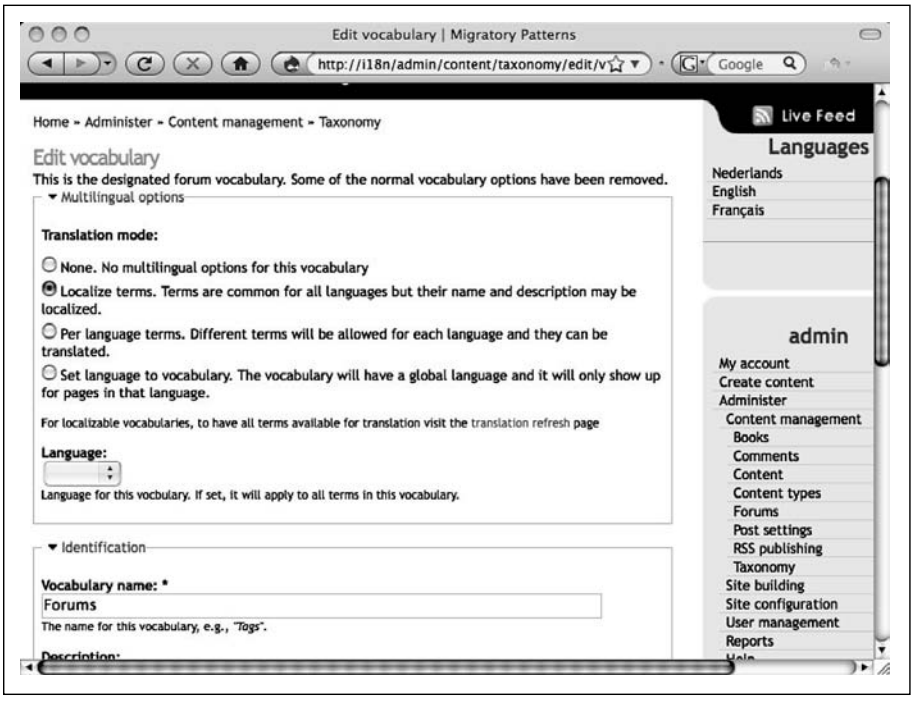


Рис. 8.34. *Параметры настройки поддержки нескольких языков для таксономии*

6. Щелкните на ссылке list terms (список терминов) для словаря Forums (Форумы) и затем щелкните на ссылке Edit (Изменить) для каждой рубрики, где вы можете изменить название и описание. Вы должны заметить в справочном тексте, что они доступны для локализации (например, This name will be localizable (Это название будет доступно для локализации)), как показано на рис. 8.35. Вы можете просто ознакомиться со страницей настройки и вернуться обратно, воспользовавшись функцией возврата на предыдущую страницу вашего броузера.
7. Выполнив эту операцию для каждой имеющейся рубрики, вернитесь в интерфейс перевода, на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Translate Interface (Переводить интерфейс) (*admin/build/translate*). Вы можете заметить, что в таблице по-прежнему отображается единственная строка, доступная для перевода. Такое может происходить при наличии некоторого содержимого. В нашем случае – это рубрики таксономии из профиля установки, которые были созданы до включения модуля Internationalization. Заставим систему Drupal выполнить повторную проверку.

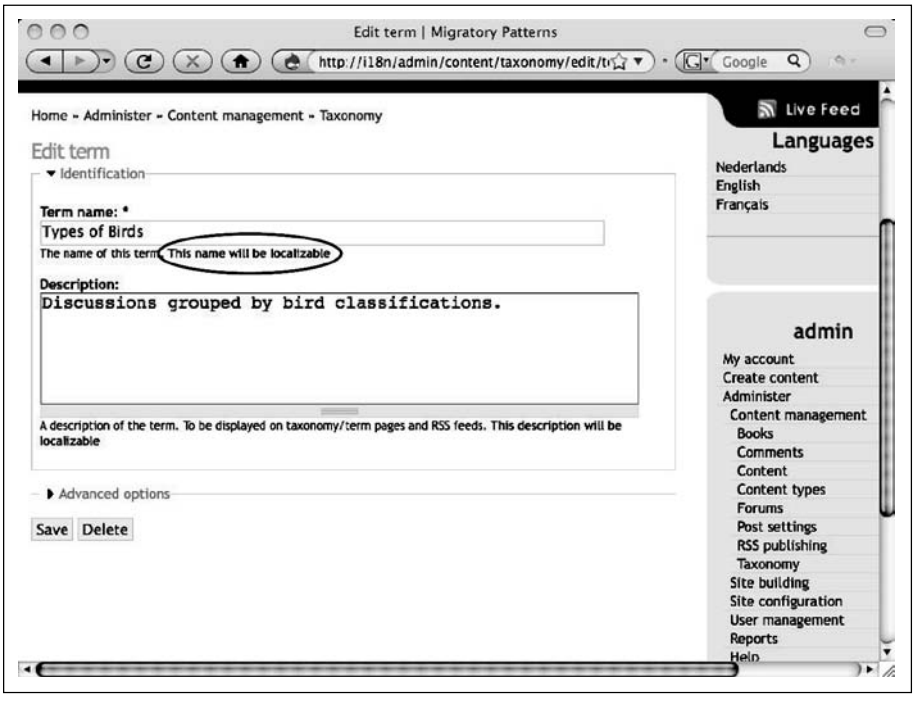


Рис. 8.35. *Справочный текст под названием рубрики указывает, что оно доступно для локализации*

8. Щелкните на вкладке Refresh (Обновить) (*admin/build/translate/refresh*), затем отметьте флажок Taxonomy (Таксономия) и щелкните на кнопке Refresh strings (Обновить строки). Страница, где можно выполнить обновление, показана на рис. 8.36.
9. Вернитесь на вкладку Overview (Обзор) (*admin/build/translate*) – мы наконец-то получили все семь строк. Почему семь? Четыре строки – это название словаря Forums, плюс три рубрики и еще три строки – описания, по одному для каждой рубрики.
10. Теперь вы можете отыскать и перевести строки, как мы это делали прежде, после чего при переключении языка будут отображаться локализованное название форума и описание.

Дополнительно

Мы рассмотрели достаточно много базовых сведений, необходимых для создания сайтов с поддержкой нескольких языков. Для работы с языками существует множество инструментов, обеспечивающих значительную гибкость. Для нашего клиента мы создали сайт, обладающий

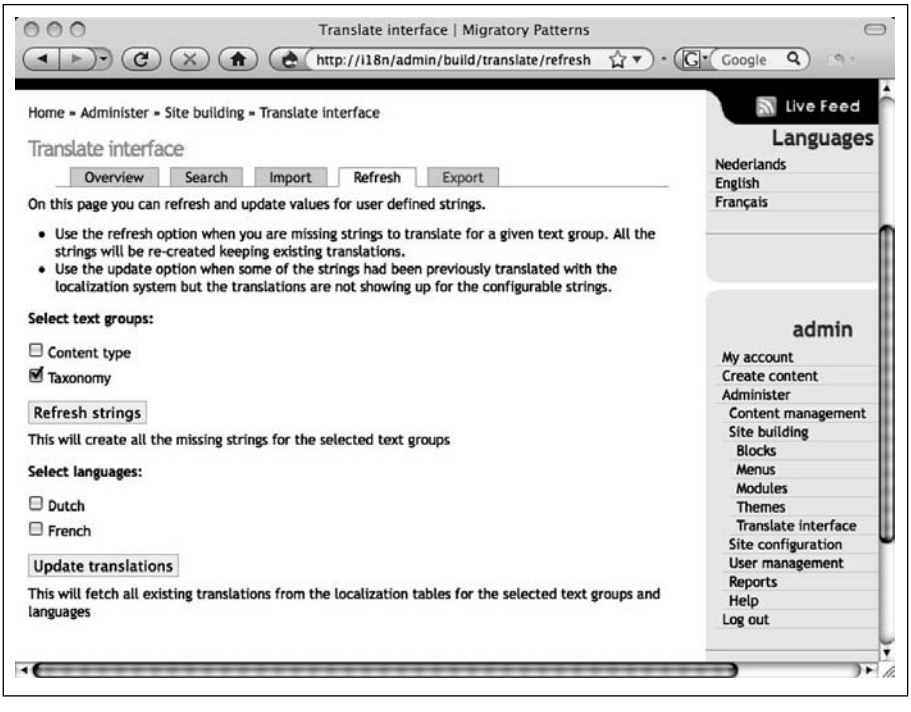


Рис. 8.36. *Страница обновления списка строк*

особенностями, необходимыми сообществу, и дающий пользователям возможность участвовать в наполнении сайта и управлять им на своем родном языке. В будущем они легко смогут добавлять новые языки, и любой желающий сможет помочь с переводом базы знаний. Ниже приводится список дополнительных модулей, которые могут придать сайту дополнительные привлекательные черты:

Language icons (http://drupal.org/project/languageicons)

Этот модуль добавляет стандартный набор ярлыков с флагами для ссылок переключения языка (для блока переключения языков и для ссылок переключения языка в содержимом). Вы можете заменить эти ярлыки своими собственными.

Translation Overview (http://drupal.org/project/translation_overview)

Этот модуль создает страницу с таблицей, содержащей информацию о состоянии перевода всего содержимого на сайте. В таблице выводится такая информация, как заголовки со ссылками на материалы, типы содержимого и дата создания. Кроме того, для каждого элемента содержимого выводится информация, взглянув на которую можно сразу оценить состояние перевода (оригинальный язык, свежий перевод, устаревший перевод или отсутствие перевода).

Auto Timezone (http://drupal.org/project/autotimezone)

Пользователи смогут устанавливать свой часовой пояс на странице My Account (Мой профиль), чтобы дата и время создания материалов отображались в локальном времени. Этот модуль включает данную возможность автоматически, опираясь на настройки броузера.

В заключение

Мы построили для нашего клиента отличный, простой в использовании сайт, который содержит все необходимое для ведения коллективных дискуссий и хранения базы знаний. Мы встроили форум, который отображает только сообщения на языке пользователя, и базу знаний, содержимое которой смогут переводить все зарегистрированные пользователи.

Главная потребность сообщества заключается в возможности использовать несколько языков и добавлять поддержку новых языков в будущем. Использовав встроенные в Drupal средства интернационализации с небольшим количеством сторонних модулей, нам удалось реализовать весьма гибкое решение.

За дополнительной информацией и для обсуждения вопросов, связанных с интернационализацией в Drupal, обращайтесь на страницу группы интернационализации, по адресу *http://groups.drupal.org/i18n*. Чтобы получить дополнительные сведения о переводах ядра Drupal и о том, как можно помочь с переводами, обращайтесь на страницу группы переводов, по адресу *http://groups.drupal.org/translations*. Если у вас появится желание помочь проекту своими переводами ядра Drupal или сторонних модулей, ознакомьтесь с руководством переводчика, по адресу *http://drupal.org/contribute/translations*, где вы найдете всю необходимую информацию.

Ниже приводится перечень модулей, упоминавшихся в этой главе:

- Auto Timezone: *http://drupal.org/project/autotimezone*
- Content Translation: входит в состав ядра Drupal
- Devel module: *http://drupal.org/project/devel*
- Internationalization и Language Icons: *http://drupal.org/project/i18n*
- Locale: входит в состав ядра Drupal
- Localization client: *http://drupal.org/project/l10n_client*
- String Overrides: *http://drupal.org/project/stringoverrides*
- Translation Overview: *http://drupal.org/project/translation_overview*

Ниже приводится список ресурсов, упоминавшихся в этой главе, и ресурсов сообщества для более подробного изучения новых концепций, представленных в этой главе:

- Переводы ядра Drupal: *http://drupal.org/project/translations*
- Модули поддержки многоязычности: *http://drupal.org/project/Modules/category/97*
- О переводе переменных сайта на другие языки: *http://drupal.org/node/313272*
- Рабочая группа переводов: *http://groups.drupal.org/translations*
- Руководство переводчика: *http://drupal.org/contribute/translations*

9

Управление событиями

Управление расписаниями и регистрацией событий может явиться серьезным вызовом. Без динамических механизмов эта задача практически неразрешима. Создание разметки HTML, которая отображала бы расписание и все различные варианты отображения (на день, на неделю, на месяц и так далее), представляется немыслимым; и более того, так как эти события имеют привязку ко времени, контролировать обновление списков «будущих» или «предстоящих» событий становится непосильной задачей. Трудно представить себе что-то хуже, чем события прошлой недели, отображаемые вверху списка на странице «Предстоящие события».

Но даже динамические механизмы обычно ограничивают свободу действий определенными параметрами с фиксированными вариантами значений. При этом, используя гибкость Drupal и опираясь на мощный фундамент модулей CCK и Views, вы можете реализовать на сайте списки событий практически любых видов.

Если вы собираетесь следовать за практическими упражнениями этой главы, вам следует установить Drupal, используя профиль установки Events. В окончательном виде веб-сайт будет выглядеть, как показано на рис. 9.1 и по адресу *http://events.usingdrupal.com*. За дополнительной информацией об использовании профилей установки обращайтесь к разделу «Вступление».

В этой главе будут представлены следующие модули:

Date (http://drupal.org/project/date)

Предоставляет поле CCK для ввода информации о дате, а также ряд библиотек для выполнения операций со временем, таких как преобразование поясного времени

Calendar (http://drupal.org/project/calendar)

Представление для отображения списков материалов сайта в виде привлекательного календаря

Flag (http://drupal.org/project/flag)

Гибкий модуль, позволяющий администраторам добавлять переключатели «включено/выключено» к таким элементам, как узлы и комментарии

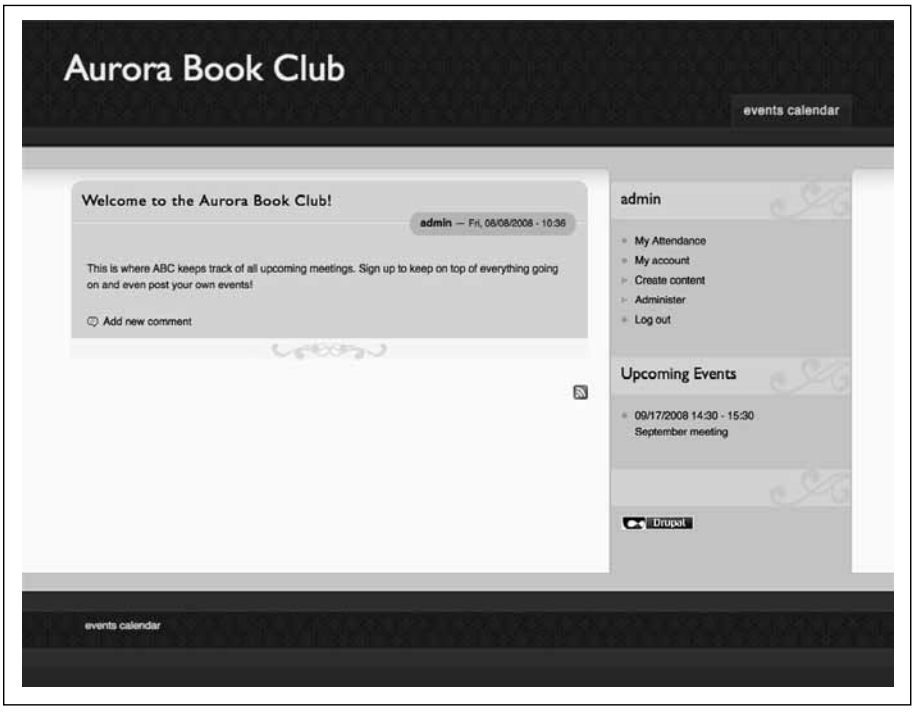


Рис. 9.1. *Окончательный вид сайта Aurora Book Club*

Задача

Aurora Book Club – это общество местных книголюбов. Они проводят ежемесячные встречи и регулярно устраивают различные мероприятия, как для действительных, так и для потенциальных членов общества. Членам общества хотелось бы иметь возможность получать информацию о времени и месте проведения намечающихся встреч. Кроме того, членам общества должна быть предоставлена возможность оставлять на сайте свои предложения о мероприятиях. Событие должно иметь дату, время начала и конца, а также описание события и место, где оно будет происходить. Чтобы легче было знакомиться с будущими событиями, в дополнение к полному расписанию на сайте должен присутствовать краткий список наступающих событий. Расписание должно позволять просматривать события за день, за месяц и за год, а для

членов общества еще и предоставлять возможность подписываться на события в календаре клуба с помощью Microsoft Outlook или Apple iCal. Наконец, президент общества попросил добавить возможность следить за численностью подписавшихся на посещение каждого события, чтобы знать, сколько угощенья и сколько стульев следует предоставить.

Примечания к реализации

Управление событиями

Наше общество книголюбов может полагаться на два основных варианта управления событиями в системе Drupal. Эти варианты отражают глобальные тенденции развития сторонних модулей для Drupal. С одной стороны, имеется давно устоявшийся модуль Drupal, который называется просто Event (*http://drupal.org/project/event*). Он обладает практически всеми необходимыми нам особенностями. Второй вариант, который мы и применим в этой главе, – это использование дополнений к пакетам CCK и Views: модуль Date (*http://drupal.org/project/date*) и модуль Calendar (*http://drupal.org/project/calendar*), соответственно. Ниже приводятся основные отличия между этими вариантами:

- Модуль Event позволяет включать поддержку событий в типы содержимого, то есть добавлять время начала и конца события к любому существующему типу содержимого. В состав модуля также входит все необходимое для просмотра созданных событий и даже блок для вывода списка наступающих событий.
- Модуль Date предоставляет поле CCK для работы с датами. Модуль Calendar – это расширение для модуля Views, отображающее представление в виде календаря.

Оба способа в состоянии удовлетворить все требования, предъявляемые к сайту общества книголюбов. В действительности, если сравнивать их между собой, функционально они выглядят эквивалентными: они оба дают возможность создавать новые события, просматривать списки предстоящих событий и отображать расписание в виде календаря с широкими функциональными возможностями. На самом деле, при работе с любыми типами содержимого модуль Event также позволяет использовать модуль CCK при создании нестандартных типов содержимого для наших событий (например, для хранения дополнительной информации о событиях). Это обстоятельство существенно размывает границу между решениями, особенно в сравнении с выбором между модулями Image и Imagefield, который мы делали в главе 7. Модуль Image определяет конкретный тип содержимого («изображение»), тогда как модуль Event способен оперировать любым нестандартным типом содержимого. Более того, Event – это отдельный модуль, не имеющий никаких зависимостей, тогда как модули Date и Calendar требуют

установки модулей CCK и Views, соответственно. Тогда почему бы нам не использовать модуль Event?

Как уже говорилось в нескольких предыдущих главах, модули CCK и Views соответствуют перспективам в конструировании сайтов на основе Drupal. Их гибкость и избирательность обеспечивают для включаемых модулей намного более широкие возможности в точности удовлетворять потребности сайтов. Будучи узкоспециализированными модулями, Date и Calendar также более полно реализуют свои специфические особенности. Это особенно справедливо для модуля Date, который обладает более широкими возможностями форматирования дат, чем Event. Кроме того, опираясь на решения, основанные на модулях CCK и Views, мы будем автоматически включены в получение всех улучшений и новых преимуществ, появляющихся по мере развития модулей CCK и Views.

По сути, применение модуля Event обычно обусловлено либо необходимостью сохранения единства с уже имеющимися наработками, либо простотой установки. За дополнительной информацией о модулях, связанных с управлением событиями, обращайтесь в раздел «Event» на сайте Drupal.org (*http://drupal.org/project/Modules/category/61*).

Слежение за численностью участников

Модуль Signup *http://drupal.org/project/signup* проектировался специально для целей слежения за численностью тех, кто выразил желание посетить мероприятие, и обладает такими замечательными возможностями, как рассылка по электронной почте напоминаний о предстоящем событии. Однако этот модуль недоступен в версии Drupal 6 и к моменту написания этих строк все еще находился на стадии разработки.

Поэтому мы откажемся от него и воспользуемся случаем продемонстрировать полезный, универсальный модуль Flag (*http://drupal.org/project/flag*). Модуль Flag (в версии Drupal 5 известный как Views Bookmark) дает пользователям возможность помечать или устанавливать «флаги» на элементы содержимого. Эту особенность можно использовать в самых разных случаях, включая маркировку содержимого как оскорбительного, позволять пользователям устанавливать закладки на интересные материалы и даже отмечать события, как «стоящие посещения» или «не стоящие посещения».

Практика: первые шаги

В первую очередь нам необходимо создать фундамент нашего сайта, задействовав ядро Drupal и CCK. Самое первое, что нужно сделать, – это определить тип содержимого для работы с событиями. Если вы использовали профиль установки, зарегистрируйтесь на сайте Aurora Book Club как пользователь *admin* с паролем *oreilly*.

Создание типа содержимого Event

Начнем с создания нового, базового типа содержимого для событий. Для каждого события нам необходимо иметь возможность определить его название, описание и место:

1. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Modules (Модули) (*admin/build/modules*) и включите следующие модули:
   - CCK
     – Content
     – Text
2. Перейдите на страницу Administer (Администрирование)→Content management (Содержание)→Content types (Типы материалов) (*admin/content/types*) и щелкните на вкладке Add content type (Добавить тип содержимого) (*admin/content/types/add*), чтобы создать новый тип содержимого с именем Event (Событие) в соответствии с табл. 9.1.

Таблица 9.1. Настройки типа содержимого Event (Событие)

Параметр	Значение
Identification (Идентификация)	
Name (Имя)	Event (Событие)
Type (Тип)	event
Description (Описание)	A book club meeting or social event (Встреча членов клуба книголюбов или общественное мероприятие)
Submission form settings (Свойства формы)	
Title field label (Название поля заголовка)	Name (Название)
Body field label (Название поля текста)	Description (Описание)
Workflow settings (Процесс)	
Default options (Настройки по умолчанию)	снять флажок Promoted to front page (Помещать на главную)
Comment settings (Установки комментариев)	
Default comment settings (Установки комментариев по умолчанию)	выключено

3. Щелкните на кнопке Save content type (Сохранить тип содержимого).
4. Добавьте текстовое поле для хранения адреса места проведения мероприятия (то есть место события). После сохранения нового типа вы должны были вернуться на страницу Content types (Типы материалов) (*admin/content/types*). Щелкните на ссылке manage fields (управлять полями) для типа содержимого Event (Событие) (*admin/content/node-type/event/fields*). Заполните форму New field (Новое поле) в соответствии с табл. 9.2.

Таблица 9.2. Настройки для поля, хранящего место проведения в содержимом типа Event (Событие)

Параметр	Значение
Label (Название)	Location (Место)
Field name (Имя поля)	location
Select a field type (Выберите тип поля)	Text (Текст)
Select a widget type (Выберите тип виджета)	Text field (Текстовое поле)

5. Щелкните на кнопке Save (Сохранить). После этого вы окажетесь на странице настройки поля Location (Место). Здесь оставьте все настройки со значениями по умолчанию и просто щелкните на кнопке Save field settings (Сохранить настройки поля).

Права доступа

Теперь, после того как новый тип содержимого был создан и настроен, необходимо дать членам клуба разрешение на его использование, чтобы они могли создавать свои события.

На странице Administer (Администрирование)→User management (Управление пользователями)→Permissions (Разрешения) (*admin/user/permissions*) установите разрешения в соответствии с табл. 9.3. По окончании щелкните на кнопке Save permissions (Сохранить права доступа).

Таблица 9.3. Разрешения для типа содержимого Event (Событие)

Разрешение	anonymous user (анонимный пользователь)	authenticated user (зарегистрированный пользователь)	editor (редактор)	site administrator (администратор сайта)
модуль node				
create event content (создавать материал типа event)		отмечен		

delete any event content (удалять любой материал типа event)				отмечен
edit any event content (изменять любой материал типа event)			отмечен	отмечен
edit own event content (изменять собственный материал типа event)		отмечен		

В центре внимания: модуль Date

Основным строительным блоком нашего сайта является наш новый тип содержимого Event (Событие). Вместе с событиями мы должны передавать такую информацию, как «где» и «когда». Об информации «где» мы уже побеспокоились на начальном этапе. Модуль Date поможет нам ответить на вопрос «когда», обеспечивая при этом необычайную гибкость.

Как уже упоминалось выше, наша истинная заинтересованность в модуле Date обусловлена необходимостью добавить в тип содержимого Event (Событие) поле CCK, которое хранило бы дату и время. Однако при более пристальном рассмотрении модуля Date можно заметить наличие некоторых дополнительных компонентов, о которых следует рассказать.

Модуль Date API

Как следует из его имени, модуль Date API просто предоставляет комплект библиотечных функций для работы с датами. Среди них присутствуют функции преобразования между форматами, обеспечения интеграции с модулями, например с Views, и даже функции для создания полей ввода дат. Хотя полный охват прикладного интерфейса выходит далеко за рамки этой главы, тем не менее следует отметить, что модуль Date API не зависит от CCK, который требуется модулям Date и Calendar. Теоретически любой модуль Drupal, выполняющий какие-либо операции с датами, может использовать этот модуль.

Пользователи, использующие PHP4, должны задействовать модуль Date PHP4, реализующий функции работы с датами, появившиеся в версии PHP 5.2. При необходимости этот модуль может использовать внешнюю библиотеку для разбора дат в дополнительных форматах: библиотеку ADOdb date (*http://phplens.com/phpeverywhere/adodb_date_library*). Без этой библиотеки модуль Date PHP4 сможет работать только с датами, находящи-

мися в диапазоне от 1970 года по 2038 год. Библиотека добавляет поддержку любых дат в диапазоне от 100 года по 3000 год. Для нужд нашего общества книголюбов нам достаточно будет собственных возможностей модуля Date API.

Модуль Date Timezone

По умолчанию ядро Drupal интерпретирует часовые пояса, как смещение относительно Гринвичского времени (Greenwich Mean Time, GMT), которое представлено в виде разницы (плюс или минус) локального времени с Гринвичским. Например, смещение –0400 – это восточное летнее время (Eastern Daylight Time) в Соединенных Штатах. Такой подход имеет несколько недостатков, наиболее существенным из которых является необходимость учесть летнее время, в виде увеличения или уменьшения текущего смещения. С переходом на зимнее время, когда локальное время становится восточным стандартным временем (Eastern Standard Time), фактическое смещение относительно Гринвичского времени изменяется с –0400 на –0500, и возникает необходимость изменить настройки.

Модуль Date API распространяется вместе с модулем Date Timezone, который позволяет заменить смещения названиями часовых поясов (например, вместо –0400 использовать название «America/New York»). Благодаря этому модуль Date API может аккуратно и самостоятельно учитывать переход на летнее время и обратно.

Типы полей для хранения дат

На самом нижнем уровне модуль Date определяет три поля CCK – Date, Datestamp и Datetime, которые можно добавлять в типы содержимого в виде полей с датами. Различия между этими типами полей приводятся в табл. 9.4.

Таблица 9.4. Поля CCK, предлагаемые модулем Date

Имя	Описание	Пример	Тип поля в базе данных
Date	Сохраняет дату в базе данных в формате ISO8601; используется для хранения исторических дат (до 1000 года н.э.) или для хранения фрагментов дат (например, только года, без числа и месяца). Этот тип поля использовать нежелательно, так как для сортировки и преобразования дат в этом формате требуются существенные вычислительные ресурсы.	2008-08-26T17:02:00	varchar(20)

Datestamp	Тип Datestamp сохраняется в привычном для операционной системы UNIX формате timestamp, который определяет число секунд, прошедших с 1 января 1970 года. Кроме того, этот тип позволяет хранить даты в ограниченном диапазоне (в большинстве систем от 1901 года н.э. до 2038 года н.э.), но обеспечивает высокую скорость вычисления смещений часовых поясов и сортировку в списках. Устаревший формат Datestamp поддерживается всеми базами данных.	1219770120	int(11)
Datetime	Тип Datetime сохраняется во внутреннем формате «datetime» баз данных, предназначенном для представления дат. Преимущества этого типа заключаются в том, что он позволяет использовать специализированные функции баз данных для работы с датами, включая простоту извлечения отдельных элементов даты, но при этом имеется несовместимость между базами данных в поддержке этого формата.	2008-08-26 17:12:00	datetime

Еще больше, намного больше, чем вам хотелось бы знать, доводов «за» и «против» различных форматов хранения дат можно найти в интересном обсуждении в рабочей группе Events, по адресу *http://groups.drupal.org/node/731*.

Поскольку у нас нет причин переводить сайт Aurora Book Club с MySQL на использование какой-либо другой базы данных, а все даты укладываются в «привычные» рамки, мы будем использовать тип Datetime.

Виджеты ввода дат

Помимо основных типов полей, модуль Date также определяет три виджета CCK для ввода информации о дате, которые изображены на рис. 9.2:

Список выбора

Это серия раскрывающихся списков для выбора по отдельности года, месяца, числа, часа, минуты и секунды, в зависимости от настройки точности представления в поле даты.

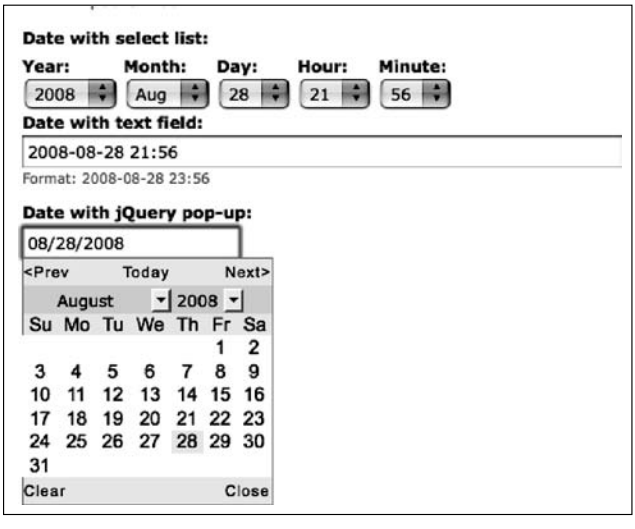


Рис. 9.2. *Виджеты CCK для ввода даты*

Текстовое поле с настраиваемым форматом ввода

Простое текстовое поле для ввода даты, которое преобразует введенный текст в соответствующий формат хранения. Преимущество этого виджета состоит в том, что он позволяет опытным пользователям вводить даты быстрее. Однако для неопытного пользователя он может превратиться в источник проблем, если виджету не удастся распознать формат введенной даты.

Текстовое поле с всплывающим календарем из библиотеки jQuery

Изящный и дружественный способ ввода дат. Этот виджет использует JavaScript для представления календаря, когда пользователь щелкает мышью на текстовом поле ввода даты. Он предоставляет возможность пользователю выбрать дату щелчком на календаре.

Для большего удобства на сайте общества книголюбов будет использоваться виджет «Текстовое поле с всплывающим календарем из библиотеки jQuery».

Настройки модуля Date

Для данного поля ввода даты существует совсем немного настроек, отличных от других типов полей CCK; они приведены на рис. 9.3 и 9.4. Ниже перечислены некоторые из них:

Default value (Значение по умолчанию)

Настройка значения по умолчанию предоставляет выбор из нескольких вариантов. Варианты `Blank` (Пустое) и `Now` (Текущее время) говорят сами за себя. Вариант `Relative` (Относительно) позволяет устанавливать дату со смещением относительно текущего времени, на-

пример через два дня от текущей даты. Чтобы выбрать по умолчанию относительное значение, необходимо ввести в поле Customize Default Value (Настраиваемое значение по умолчанию) значение в формате, который используется при вызове функции *strtotime()* в языке PHP, например: +2 дня. Подробнее с функцией *strtotime()* можно познакомиться по адресу *http://www.php.net/manual/en/function.strtotime.php*.[1]

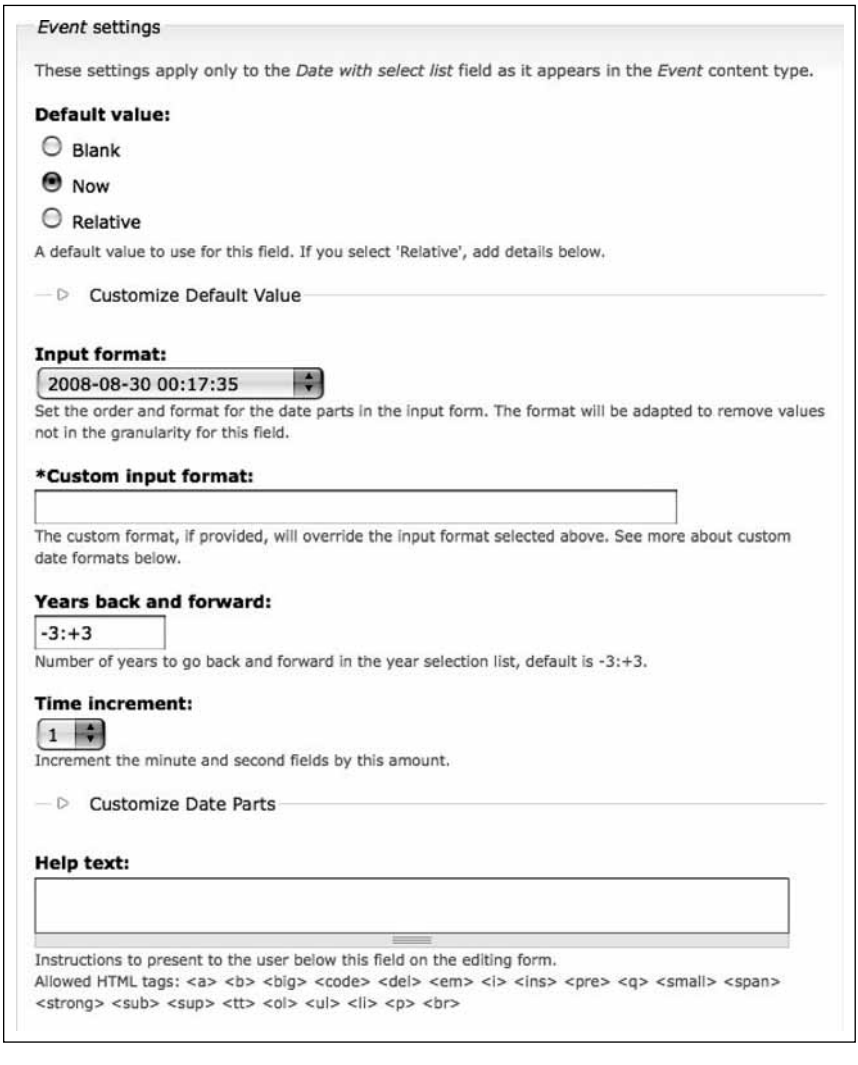


Рис. 9.3. *Настройки поля Date для типа содержимого*

[1] На русском языке *http://www.php.ru/manual/function.strtotime.html*. – *Примеч. перев.*

Global settings

These settings apply to the *Date with select list* field in every content type in which it appears.

☐ Required

Number of values:

1

Select a specific number of values for this field, or 'Unlimited' to provide an 'Add more' button so the users can add as many values as they like.

Warning! Changing this setting after data has been created could result in the loss of data!

To Date:

◉ Never

○ Optional

○ Required

Display a matching second date field as a 'To date'. If marked 'Optional' field will be presented but not required. If marked 'Required' the 'To date' will be required if the 'From date' is required or filled in.

Granularity:

Year
Month
Day
Hour
Minute
Second

Set the date elements to be stored (at least a year is required).

Default Display

Date display:

08/26/2008 - 17:01

***Custom display format:**

▷ Additional Display Settings

Time zone handling:

No time zone conversion

Select the timezone handling method to be used for this date field.

Рис. 9.4. *Настройки поля типа Date*

Input format (Формат ввода)

Настройка формата ввода определяет порядок следования элементов даты – года, месяца, числа, часа, минут и секунд – при отображении в форме ввода. При этом будет учитываться значение настройки Granularity (Точность). Кроме того, существует настройка Custom input format (Другой формат ввода), которая позволяет указывать произвольный формат ввода даты, обеспечивающий непревзойденную гибкость, – для использования в виджетах «Список выбора» и «Текстовое поле с настраиваемым форматом ввода». Собственный формат определяется с использованием синтаксиса функции *date()* в языке PHP (*http://php.net/date*)[1]. Виджет «Текстовое поле с всплывающим календарем из библиотеки jQuery» не позволяет указывать собственный формат ввода, потому что он вынужден использовать свой формат, чтобы обеспечить нормальную работу календаря.

Years back and forward (Количество лет до и после)

Этот параметр позволяет определить диапазон дат, которые будут доступны пользователю для выбора в виджетах. Пример формата: *–2:+4*, где *–2* – это количество лет, предшествующих текущему году, а *+4* – это количество лет после текущего года. Если принять, что сейчас 2009 год, то при использовании этого формата доступным будет диапазон от 2007 года (–2 года) до 2013 года (+4). Для нашего сайта нет смысла использовать более широкий диапазон, особенно если говорить о событиях в прошлом.

Time increment (Приращение времени)

Этот параметр позволяет ограничивать количество минут (и секунд, в зависимости от значения параметра Granularity (Точность)), доступных для выбора. По умолчанию для выбора доступны все 60 минут. Однако во многих случаях дата и время могут вводиться с 15- или 30-минутными интервалами. В связи с этим наличие всех 60 минут в раскрывающихся списках делает их слишком неудобными для использования.

Customize date parts (Настройка элементов даты)

Поле ввода даты состоит из нескольких полей и их названий. Имеется возможность определить, как будут располагаться подписи относительно полей: над полями, внутри поля (в виде одного из вариантов выбора в раскрывающихся списках или внутри текстового поля) или вообще не будут отображаться.

Виджет «Select list» (Список выбора) имеет чуть больше настроек. Несмотря на название, мы можем реализовать ввод некоторых элементов даты в виде текстовых полей, смешивая раскрывающиеся списки и текстовые поля. Например, вместо того, чтобы организовать ввод

[1] На русском языке *http://www.php.ru/manual/function.date.html*. – *Примеч. перев.*

числа в виде раскрывающегося списка, содержащего 31 день, можно указать, что поле Day (Число) будет текстовым полем ввода, в этом случае для ввода года и месяца будут использоваться раскрывающиеся списки, а для ввода числа – небольшое текстовое поле. Это позволяет нам управлять отображением виджета и выбирать интерфейсные элементы, наиболее удобные для целевой аудитории.

В группе Global settings (Общие настройки) присутствует еще больше настроек, связанных с особенностями обработки поля:

To Date (Конечная дата)

Параметр To Date (Конечная дата) позволяет превратить виджет ввода даты/времени в виджет ввода интервала между двумя датами. Этот параметр может иметь следующие значения:

Never (Никогда)

Конечная дата для данного поля не определена.

Optional (Необязательно)

Потенциально поле может иметь конечную дату, и виджет для ввода будет отображаться, но он может оставаться незаполненным.

Required (Обязательно)

Конечная дата должна вводиться, если поле даты является обязательным или если была введена начальная дата.

Granularity (Точность)

Параметр настройки точности определяет, как много информации о дате будет сохраняться. Флажки Year (Год), Month (Месяц), Day (Число), Hour (Часы), Minute (Минуты) и Second (Секунды) могут выбираться независимо друг от друга, что обеспечивает чрезвычайно высокую гибкость. Так, если необходимо было бы указать только дату дня рождения (но не полную дату рождения), можно было бы выбрать только флажки Month (Месяц) и Day (Число); например, 10 июля. Для нужд управления событиями вполне подходит выбор по умолчанию, включающий Year (Год), Month (Месяц), Day (Число), Hour (Часы), Minute (Минуты), что позволяет отображать дату и время, такую как 10 июля, 2008 – 7:30.

Обратите внимание, что настройка параметра Granularity (Точность) определяет вид виджета ввода, в котором будут отображаться только соответствующие поля ввода.

Default display (Отображение по умолчанию)

Группа параметров Default display (Отображение по умолчанию) позволяет настраивать формат отображения значений даты. Существует четыре формата отображения, которые мы можем настроить: по умолчанию, который можно видеть на рис. 9.4, краткий, промежуточный и полный, допускающий возможность дополнительной настройки с помощью группы параметров Additional Display Settings (До-

полнительные настройки отображения). Настройка отображения включает два параметра. Раскрывающийся список Date display (Отображение даты) содержит огромное число доступных форматов представления дат (месяц может выводиться как название и как число; время может выводиться в 12- и 24-часовом формате; имеется возможность указывать часовой пояс, а также выбирать, в каком порядке должны следовать год, месяц и число). Если ни один из имеющихся форматов не подходит по каким-либо причинам, ниже находится дополнительное текстовое поле ввода Custom display format (Другой формат отображения), позволяющее выполнять дальнейшую настройку, а также имеется настройка Custom input format (Другой формат ввода), упоминавшаяся выше.

Каждый из четырех форматов – по умолчанию, краткий, промежуточный и полный – будет доступен в виде формата CCK в настройках отображения содержимого, а также в настройках модуля Views и темы оформления.

Time zone handling (Поддержка часовых поясов)

Настройки поддержки часовых поясов позволяют определять, будут ли влиять часовые пояса на сохраняемые значения дат и нужно ли выполнять преобразования. Доступные варианты поддержки описываются в табл. 9.5.

Таблица 9.5. Варианты поддержки часовых поясов в поле типа Date

Вариант	Описание
Site's time zone (Часовой пояс на сайте)	Часовой пояс определяется для всего сайта, на странице Administer (Администрирование) > Site configuration (Настройка сайта) > Date and Time (Дата и время) (*admin/settings/date-time*). Настройку удобно использовать, когда необходимо обеспечить единство представления времени для всего сайта, даже если пользователи живут в разных часовых поясах.
Date's time zone (Часовой пояс даты)	Добавляет рядом в виджет ввода даты раскрывающийся список выбора часового пояса. Удобно для сайтов, в которых ввод дат будет осуществляться большим числом пользователей, проживающих в разных часовых поясах.
User's time zone (Часовой пояс пользователя)	Часовой пояс будет устанавливаться для каждого пользователя в отдельности, в соответствии с настройками на странице My Account (Мой профиль), если включена настройка на странице Administer (Администрирование)→Site configuration (Настройка сайта)→Date and Time (Дата и время) (*admin/settings/date-time*). Настройку удобно использовать, когда основная часть мероприятий проводится в одном часовом поясе, а пользователи проживают в разных часовых поясах.

Таблица 9.5. Варианты поддержки часовых поясов в поле типа Date (продолжение)

Вариант	Описание
UTC	Универсальное глобальное время (Coordinated Universal Time, UTC), которое фактически является эквивалентом Гринвичского времени (GMT). Это стандартный часовой пояс, который является одинаковым для всех систем.
No time zone conversion (Без преобразования часового пояса)	Используется для событий, которые определяются только датой, без указания времени, или для сайтов с локальными событиями и пользователями, когда не требуется выполнять преобразований часовых поясов.

Для сайта общества книголюбов мы не будем выполнять преобразований часовых поясов, так как все члены общества проживают в одном месте.

Практика: добавление дат

В этом разделе мы преобразуем наш тип содержимого Event (Событие), добавив в него компонент даты. Этот тип будет краеугольным камнем сайта общества книголюбов, и все остальное будет опираться на это содержимое.

Настройка модуля Date

1. На странице Administer (Администрирование)→Site building (Конструкция сайта)→Modules (Модули) (*admin/build/modules*) включите следующие модули (обратите внимание: в случае использования PHP 4 вам необходимо также включить модуль Date PHP4):
   - Date/Time
     - Date
     - Date API
     - Date Popup
     - Date Timezone
2. На странице Administer (Администрирование)→Site configuration (Конструкция сайта)→Date and Time (Дата и время) (*admin/settings/date-time*) выберите в раскрывающемся списке Default time zone (Часовой пояс по умолчанию) ваш часовой пояс. Щелкните на кнопке Save configuration (Сохранить конфигурацию), чтобы сохранить изменения.

Добавление поля даты

После включения и настройки всех необходимых модулей можно переходить к изменению типа содержимого Event (Событие), созданного ранее, и добавить в него поле даты. Обратите внимание, что при этом мы

добавим всего одно поле CCK, которое будет хранить и время начала, и время конца события:

1. На странице Administer (Администрирование)→Content management (Содержание)→Content types (Типы материалов) (*admin/content/types*) щелкните на ссылке manage fields (управлять полями) для типа содержимого Event (Событие) (*admin/content/nodetype/event/fields*). Заполните форму New field (Новое поле) значениями из табл. 9.6.

Таблица 9.6. Настройки создания поля времени для типа содержимого Event (Событие)

Параметр	Значение
Label (Название)	Time (Время)
Field name (Имя поля)	time
Select a field type (Выберите тип поля)	Datetime
Select a widget type (Выберите тип виджета)	Text field with jQuery pop-up calendar (Текстовое поле с всплывающим календарем из библиотеки jQuery)

2. Щелкните на кнопке Save (Сохранить). После этого будет выполнен переход на страницу настройки поля. Так как модуль Date по умолчанию предназначен для управления событиями, часть настроек мы оставим со значениями по умолчанию. Введите значения в соответствии с табл. 9.7 и щелкните на кнопке Save field settings (Сохранить настройки поля), чтобы завершить настройку.

Таблица 9.7. Настройки поля даты

Параметр	Значение
Event settings (Настройки событий)	
Default value (По умолчанию)	Now (Текущее время)
Input format (Формат ввода)	Выбор формата. Например: 08/29/2008–11:31pm
Years back and forward (Количество лет до и после)	–1:+2
Time increment (Шаг приращения времени)	15
Global settings (Общие настройки)	
Required (Обязательно)	отмечено
To Date (Конечная дата)	Optional (Необязательно)

Таблица 9.7. Настройки поля даты (продолжение)

Параметр	Значение
Default Date Display (Отображение по умолчанию)	Выберите формат вида 08/29/2008–11:31pm
Time zone handling (Поддержка часовых поясов)	No time zone conversion (Без преобразования часового пояса)

3. После этого вы должны вернуться на вкладку Manage fields (Управлять полями) (*admin/content/node-type/event/fields*). Измените порядок следования полей, как показано ниже, и щелкните на кнопке Save (Сохранить):
   - Name (Название)
   - Time (Время)
   - Location (Место)
   - Description (Описание)
   - Menu settings (Настройки меню)

После того как новый тип будет окончательно создан и будут выданы все необходимые разрешения, члены общества смогут оставлять на сайте сообщения о событиях! Для этого нужно перейти на страницу Create content (Создать материал)→Event (Событие) (*node/add/event*) и заполнить форму в соответствии с табл. 9.8. Если при этом не было допущено ошибок, вы должны увидеть форму, похожую на рис. 9.5. Теперь создайте на сайте Aurora Book Club еще несколько сообщений о событиях.

Таблица 9.8. Пример первого сообщения о событии

Параметр	Значение
Name (Название)	Monthly meeting (Ежемесячная встреча)
Time (Время)	
Choose a time begin (Выберите время начала)	09/17/2008–2:30PM
Choose a time end (Выберите время окончания)	09/17/2008–3:30PM
Location (Место)	The Book Nook on Main Street (В книжном уголке на Главной улице)
Description	Andrew and Camryn are bringing cookies. (Эндрю и Камрин приносят печенье.)

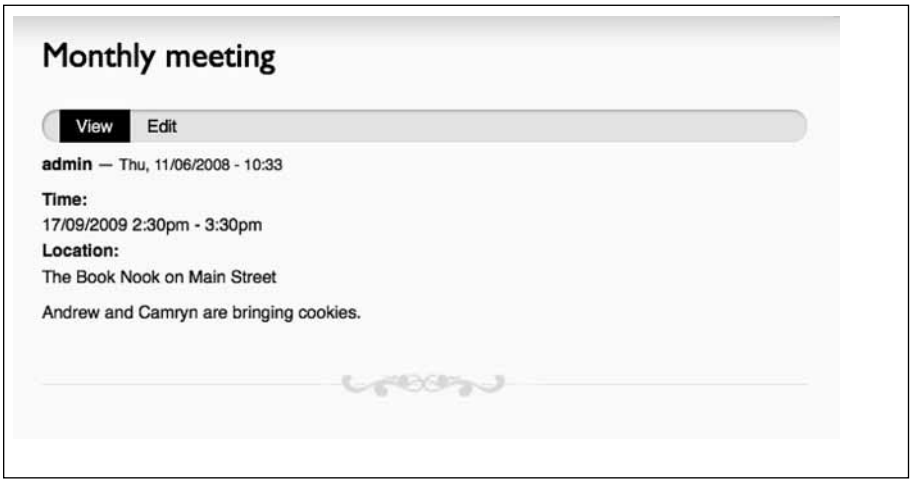


Рис. 9.5. *Наше первое сообщение о мероприятии*

Практика: представление для отображения предстоящих событий

Теперь, когда у нас имеется тип содержимого для событий и несколько сообщений этого типа, очевидно, что нам необходимо добавить возможность доступа к информации о событиях. В требованиях общества книголюбов указано, что на сайте должен иметься список «предстоящих событий», который позволил бы членам общества быстро увидеть, какие встречи намечены на ближайшие дни или недели. Для реализации такого списка мы воспользуемся модулем Views, с помощью которого создадим блок. Следует учесть, что при создании представлений для отображения информации о событиях обычно бывает необходимо выполнять сортировку или ограничивать отображаемые события по полю даты, а не по времени создания или изменения содержимого, как это обычно принято.

Мы создадим простое представление типа Block (Блок) отображения опубликованных событий, для которых время события находится в будущем. В терминах настройки представления «в будущем» означает, что значение даты события «больше текущего». Наконец, представление будет сортировать события в хронологическом порядке (по возрастанию) по дате события (а не по времени создания сообщения о событии). По окончании этого раздела главная страница сайта будет выглядеть, как показано на рис. 9.6. Щелчок мышью на названии события в блоке будет открывать страницу с полной информацией о нем.

1. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Modules (*admin/build/modules*) и включите следующие модули:

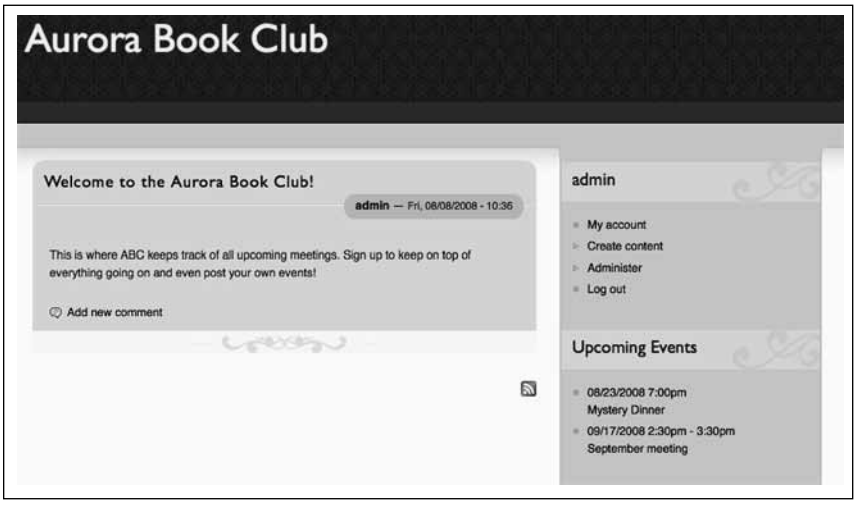


Рис. 9.6. *Список предстоящих событий на сайте Aurora Book Club*

- Other (Другой)
  - Advanced help
- Views
  - Views
  - Views UI

2. После включения модулей перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Views (*admin/build/views*) и щелкните на вкладке Add (Добавить) (*admin/build/views/add*).
3. Заполните форму настройки в соответствии с табл. 9.9.

Таблица 9.9. Настройки представления Upcoming Events (Предстоящие события)

Параметры настройки представления	**Значение**
View name (Название представления)	upcoming_events
View description (Описание представления)	A block list of upcoming events (Блок со списком предстоящих событий)
View tag (Рубрика представления)	event
View type (Тип представления)	Node (Материал)

4. Щелчок на кнопке Next (Далее) откроет страницу с основным интерфейсом настройки модуля Views. Здесь мы можем настроить наше представление. Начнем с настройки базовых параметров отображе-

ния Defaults (По умолчанию) в соответствии с табл. 9.10. Каждый раз после изменения параметра не забывайте щелкать на кнопке Update (Обновить).

Таблица 9.10. Базовые настройки представления Upcoming Events (Предстоящие события)

Defaults: Basic settings (По умолчанию: Базовые настройки)	**Значение**
Title (Название)	Upcoming Events (Предстоящие события)
Style (Стиль)	Выберите стиль List (Список) и щелкните на кнопке Update (Обновить) List type (Тип списка): Unordered list (Ненумерованный список)
Items to display (Количество отображаемых элементов)	Defaults (По умолчанию): Items to display (Количество отображаемых элементов): 5

5. Теперь необходимо добавить некоторые поля в представление. В разделе Fields (Поля) щелкните на ярлыке с изображением знака «плюс» и добавьте следующие поля: Content: Time (field_time value) (Содержимое: Время (значение field_time)) и Node: Title (Узел: Заголовок). Выполните настройки в соответствии с табл. 9.11, не забывая щелкать на кнопке Update (Обновить) после изменения каждого значения.

Таблица 9.11. Значения полей представления Upcoming Events (Предстоящие события)

Defaults: Fields (По умолчанию: Поля)	**Значение**
Content: Time (field_time value) (Содержимое: Время (значение field_time))	Label (Название): пусто
Node: Title (Узел: Заголовок)	Label (Название): оставить пустым Link this field to its node (Связать это поле с указанным узлом): отмечено

6. В разделе Filters (Фильтры) добавьте необходимые фильтры, щелкнув на ярлыке с изображением знака «плюс»:
   - Date: Date (Дата: Дата)
   - Node: Published (Узел: Опубликовано)
   - Node: Type (Узел: Тип)

7. Вспомните, что наше представление предназначено для вывода списка предстоящих (то есть будущих) событий; поэтому мы должны оставить только те элементы списка, в которых указана дата больше текущей. Установите настройки фильтра в соответствии с табл. 9.12 и обновите их.

Таблица 9.12. Значения фильтра для представления Upcoming Events (Предстоящие события)

Defaults: Filters (По умолчанию: Фильтры)	**Значение**
Date: Date (Дата: Дата)	Выберите Content: Time (field_time_value) (Содержимое: Время (значение field_time)) в поле Date field(s) (Поле даты) и щелкните на кнопке Update (Обновить) Operator (Оператор): Is greater than or equal to (Больше или равно) Date default (Дата по умолчанию): now (текущее время)
Node: Published (Узел: Опубликовано)	Published (Опубликовано): отмечено
Node: Type (Узел: Тип)	Operator (Оператор): Is one of (Если один из) Node type (Тип узла): Event (Событие)

8. Затем нам необходимо также добавить критерий сортировки, щелкнув на ярлыке с изображением знака «плюс» в разделе Sort Criteria (Критерий сортировки). Выберите Content: Time (field_time value) (Содержимое: Время (значение field_time)), щелкните на кнопке Add (Добавить), установите параметр Sort order (Порядок сортировки) в значение Ascending (По возрастанию) и щелкните на кнопке Update (Обновить) по завершении.
9. Чтобы создать блок, добавьте отображение Block (Блок), выбрав элемент Block (Блок) в раскрывающемся списке, который расположен слева в интерфейсе, и щелкните на кнопке Add display (Добавить отображение).
10. В разделе Block settings (Настройки блока) в группе Admin (Административный раздел) введите в параметр Block: Block admin description (Блок: Описание блока в административном разделе) описание «Upcoming Events» (Предстоящие события) и щелкните на кнопке Update (Обновить).
11. Сохраните представление. Теперь страница с настройками должна выглядеть, как показано на рис. 9.7.
12. Мы создали представление в виде блока, поэтому мы не должны видеть изменения на сайте, пока не включим только что созданный

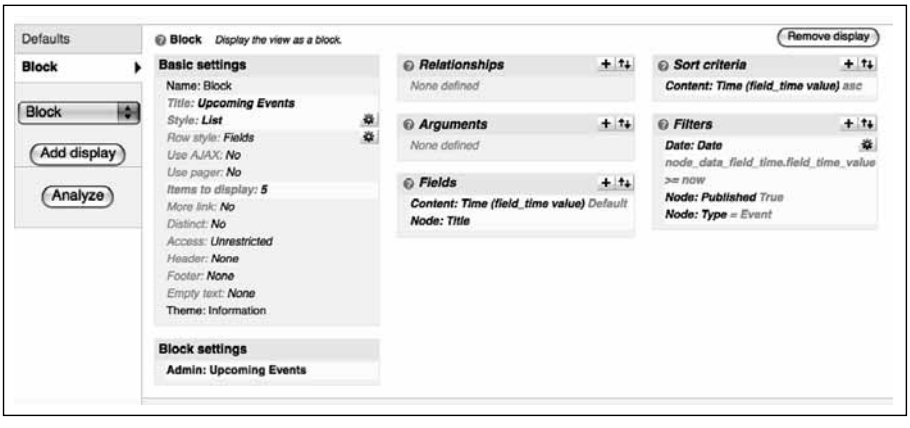


Рис. 9.7. *Страница с настройками представления Upcoming Events (Предстоящие события)*

блок. Для этого перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Blocks (Блоки) (*admin/build/block*). Отбуксируйте мышью строку «Upcoming Events» (Предстоящие события) в область Right sidebar (Правая колонка) (или просто выберите название области в раскрывающемся списке) и щелкните на кнопке Save blocks (Сохранить блоки).

В центре внимания: модуль Calendar

Наличие списка предстоящих событий – необходимая особенность для нашего сайта (особенно, когда список выводится в боковой колонке), однако у общества книголюбов имеется дополнительное требование, касающееся отображения информации о событиях. На этом сайте необходимо реализовать интерактивное расписание, которое очень часто можно встретить на сайтах, обеспечивающих возможность управления событиями, позволяющее просматривать как прошедшие, так и будущие события. Мы создадим такое расписание, воспользовавшись модулем Calendar и объединив его с модулем Views.

Помимо создания привлекательного расписания на веб-странице, модуль Calendar способен удовлетворить потребность членов общества в возможности следить за событиями с помощью своих собственных программ-органайзеров (таких как Microsoft Outlook или Apple iCal) и получать информацию о событиях с сайта общества книголюбов. Для обмена данными настольные приложения используют стандартный формат, известный как iCalendar (*http://en.wikipedia.org/wiki/iCalendar*), или, сокращенно, iCal. Вместе с модулем Calendar распространяется модуль Calendar iCal, который позволяет легко организовать обмен данными с заинтересованными пользователями в этом формате.

Тип представления Calendar

Модуль Calendar реализует представление нового типа, которое отображает информацию не в виде списка или таблицы, как представления по умолчанию, а в виде календаря. Это одно из наиболее сложных представлений. Оно обеспечивает возможность просмотра информации о событиях за день, за неделю, за месяц и за год, с большим числом ссылок на другие представления и с возможностью постраничного просмотра по дням, месяцам и годам. Для обеспечения таких богатых функциональных возможностей модуль Calendar требует наличия некоторых аргументов представления, которые должны определяться и передаваться в определенном порядке.

На основе полученных аргументов представление типа Calendar определяет, какая информация должна быть представлена пользователю. Например, если предположить, что адресом URL нашего представления будет строка *calendar*, то представление Calendar будет обрабатывать элементы пути, как описано в табл. 9.13.

Таблица 9.13. Отображение расписания в зависимости от пути

Путь	Отображается
calendar	Расписание за месяц. По умолчанию – за текущий месяц.
calendar/1970	Расписание за год. В данном случае – за 1970 год.
calendar/1970-1	Расписание за месяц. В данном случае – за январь 1970 года.
calendar/1970-1-1	Расписание за день. В данном случае – за 1 января 1970 года.

Практика: представление Calendar

В этом разделе мы подключим представление Calendar для просмотра событий на сайте общества книголюбов. Это довольно трудоемкая задача, однако модуль Calendar предлагает удобное представление по умолчанию, которое разрешает большую часть этих сложностей. В этом разделе мы настроим это представление по умолчанию под наши требования.

На рис. 9.8 показан окончательный внешний вид расписания на сайте Aurora Book Club. Обратите внимание на маленький ярлык iCal в правом нижнем углу. Щелчком на этой ссылке можно загрузить расписание в соответствующее настольное приложение.

1. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Modules (Модули) (*admin/build/module*) и включите следующие модули:

- Date/Time
  - Calendar
  - Calendar iCal

2. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Views (*admin/build/views*). Вы должны увидеть в списке пункт Default Node view: calendar (Представление узла по умолчанию: календарь). Включите его, щелкнув на ссылке Enable (Включить) (*admin/build/views/enable/calendar*).
3. Щелкните на ссылке Edit (Изменить) (*admin/build/views/edit/calendar*), чтобы открыть представление для изменения.
4. Нам необходимо изменить значения по умолчанию в разделе Arguments (Аргументы). По умолчанию при помещении события в расписание представление календаря использует дату последнего изменения узла. В нижней части в разделе Date field(s) (Поле даты) щелкните на ссылке Date: Date (Дата: Дата), а в группе параметров Date field(s) (Поле даты) снимите флажок Node: Updated date (Узел: Дата изменения) и установите флажок Content: Time (field_time value) (Содержимое: Время (значение field_time)). Затем щелкните на кнопке Update (Обновить).

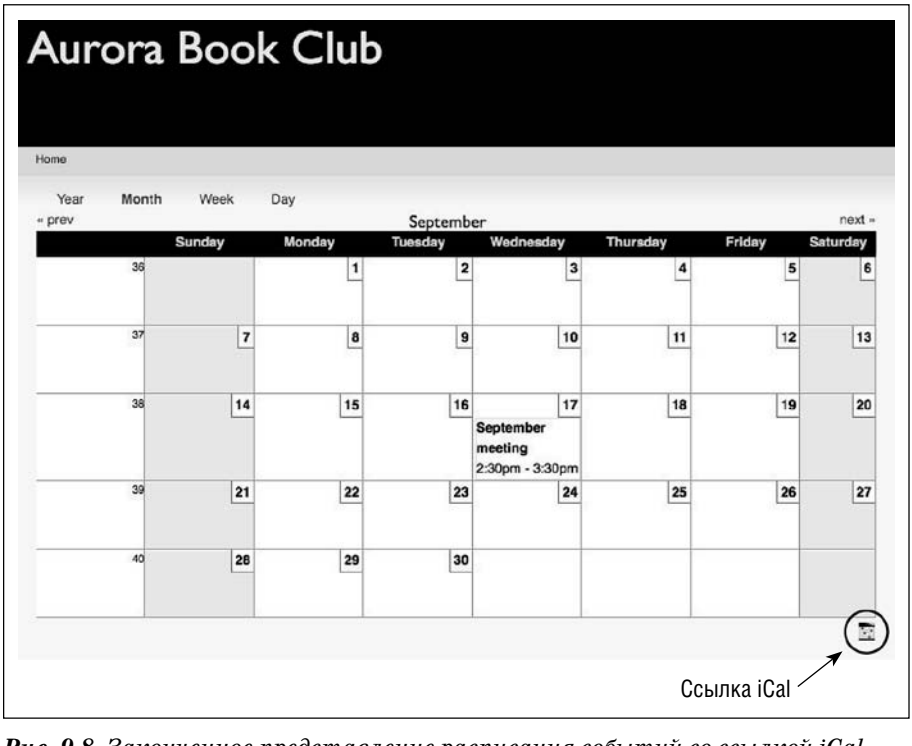


Рис. 9.8. *Законченное представление расписания событий со ссылкой iCal*

5. В разделе Fields (Поля) добавьте поле Content: Time (field_time_value) (Содержимое: Время (значение field_time)), внесите изменения в соответствии с табл. 9.14 и щелкните на кнопке Update (Обновить). Благодаря этому события будут отображаться по дате проведения, а не по дате последнего изменения узла, и из названия события будет удалено слово «Title» (Заголовок), которое выводится по умолчанию.

Таблица 9.14. Параметры настройки полей для представления Calendar

Defaults: Fields (По умолчанию: Поля)	**Значение**
Node: Updated date (Узел: Дата изменения)	(Удалить это поле)
Content: Time (field_time value) (Содержимое: Время (значение field_time))	Label (Название): пусто
Node: Title (Узел: Заголовок)	Label (Название): оставить пустым

6. Нам необходимо добавить несколько фильтров, чтобы в расписании отображались только опубликованные материалы типа Event (Событие). Для этого щелкните на ярлыке с изображением знака плюс в разделе Filters (Фильтры) и выберите фильтры Node: Published (Узел: Опубликовано) и Node: Type (Узел: Тип). Настройте фильтры в соответствии с табл. 9.15, щелкая на кнопке Update (Обновить) после каждого изменения.

Таблица 9.15. Настройки фильтров для представления Calendar

Defaults: Filters (По умолчанию: Фильтры)	**Значение**
Node: Published (Узел: Опубликовано)	Published (Опубликовано): отмечено
Node: Type (Узел: Тип)	Operator (Оператор): Is one of (Если один из) Node type (Тип узла): Event (Событие)

7. Наконец, нам необходимо добавить пункт меню, чтобы иметь возможность открывать страницу расписания. Представление по умолчанию уже предоставляет путь *calendar*, и у нас нет никаких причин менять его. Чтобы добавить его в меню, щелкните на вкладке Calendar Page (Страница расписания) в левой части интерфейса и заполните

раздел Page settings (Настройки страницы) в соответствии с табл. 9.16. После ввода настроек меню щелкните на кнопке Update (Обновить).

Таблица 9.16. Настройки страницы представления Calendar

Page: Page settings (Страница: Настройки страницы)	**Значение**
Menu (Меню)	Type (Тип): Normal menu entry (Обычный элемент меню) Title (Заголовок): Events calendar (Расписание событий)

8. Сохраните представление, настройки которого теперь должны выглядеть, как показано на рис. 9.9.

Рис. 9.9. *Настройки представления расписания Upcoming Events (Предстоящие события)*

9. Теперь можно переместить пункт меню Events Calendar (Расписание событий) в меню Primary Links (Основные ссылки). Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Menus (Меню)→Navigation (Навигация) (*admin/build/menucustomize/navigation*) и щелкните на ссылке edit (изменить) в строке с расписанием Events (События).
10. Измените параметр Parent item (Родительский пункт), выбрав значение <Primary links> (<Основные ссылки>), и щелкните на кнопке Save (Сохранить).

Теперь у нас имеется действующее расписание событий, которое можно увидеть, щелкнув на ссылке Events calendar (Расписание событий) в навигационном меню.

В центре внимания: модуль Flag

Модуль Flag (ранее известный под названием Views Bookmark) – это чрезвычайно гибкий модуль, позволяющий создавать взаимосвязи между пользователями и содержимым сайта. После создания флага элемент содержимого может быть помечен флагом разными способами, включая ссылки, отображаемые под содержимым, как показано на рис. 9.10, или флажки, отображаемые в форме редактирования, как показано на рис. 9.11.

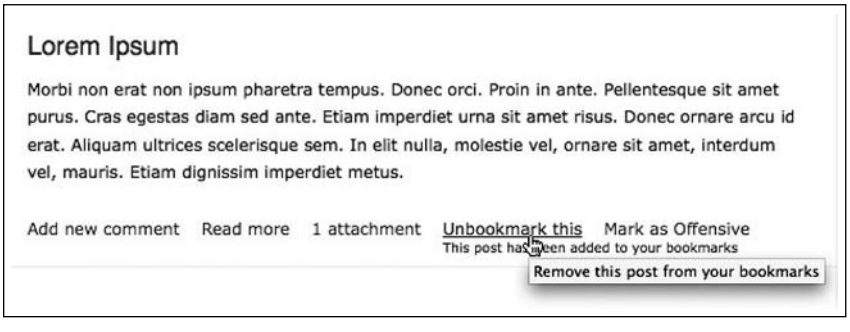


Рис. 9.10. *Флаги в виде ссылок, отображаемых под содержимым*

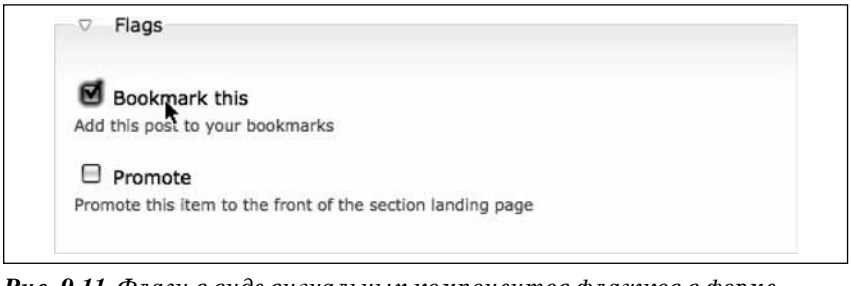


Рис. 9.11. *Флаги в виде визуальных компонентов флажков в форме редактирования материала*

После установки модуль Flag определяет тип взаимосвязей «закладка» и обеспечивает пользователей возможностью управлять списком закладок (или материалов, которые они сочли интересными для себя) на сайте. Однако такое поведение по умолчанию – лишь видимая сторона гибкости модуля Flag. Среди возможных вариантов использования модуля Flag могут быть названы:

- Маркировка содержимого такими флагами, как «favorite» («избранное») или «bookmark» («закладка»)
- Маркировка содержимого флагом «promote» («поддержать») (или несколькими такими флагами), напоминающим по своему действию флажок Promote to front page (Поместить на главную)

- Маркировка комментариев и материалов флагом «offensive» («оскорбительный»)
- Флаг «friend» («друг») дает пользователям возможность отмечать других пользователей как своих друзей
- Маркировка событий флагом «attending» («участие»)

Как видите, существуют различные способы маркировки содержимого. После создания флага для определенных целей можно сконструировать представление, создающее список материалов, помеченных пользователями. Мы воспользуемся этой возможностью, чтобы дать пользователям возможность указывать, собираются ли они принимать участие в намеченных мероприятиях общества книголюбов. После создания взаимосвязи для флага мы сможем создать представление, перечисляющее всех желающих принять участие в том или ином мероприятии.

Практика: настройка модуля Flag

В этом разделе мы выполним настройку модуля Flag с целью дать пользователям возможность указывать, что они намерены принять участие в том или ином мероприятии, как показано на рис. 9.12.

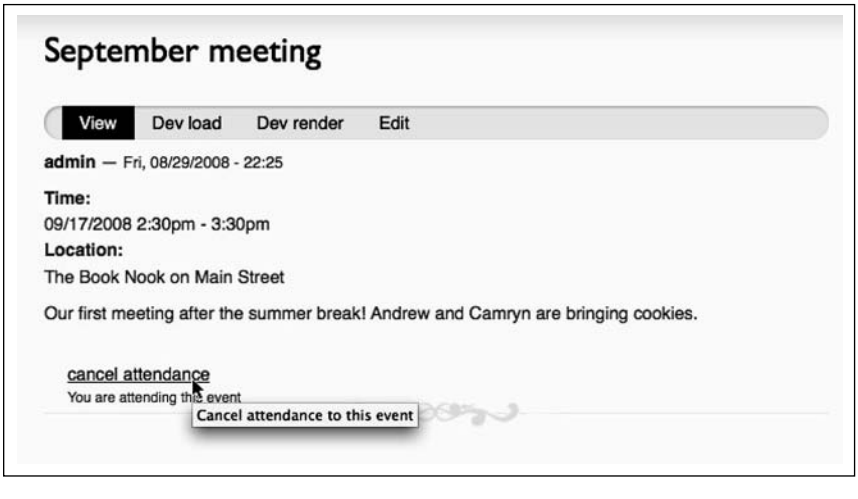


Рис. 9.12. *Признак участия в мероприятии, отображаемый в форме*

1. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Modules (Модули) (*admin/build/modules*) и включите модуль Other: Flag (Другой: Flag).
2. Теперь перейдите на страницу управления флагами: Administer (Администрирование)→Site building (Конструкция сайта)→Flags (Флаги) (*admin/build/flags*). По умолчанию модуль Flag определяет един-

ственный флаг «Bookmarks» (Закладки). Мы можем либо добавить новый флаг, либо изменить флаг по умолчанию, приспособив его под свои цели. Так как мы не собираемся использовать флаг «Bookmarks» (Закладки), мы изменим флаг по умолчанию. Щелкните на ссылке edit (изменить) рядом с флагом bookmark (закладка) (*admin/build/flags/edit/bookmarks*). Заполните форму в соответствии с табл. 9.17.

Таблица 9.17. Настройки флага attendance (участие)

Параметр	Значение
Name (Имя)	attendance (участие)
Title (Заголовок)	Attendance (Участие)
Flag link text (Текст ссылки для установки флага)	attend this event (Участвовать в этом мероприятии)
Flag link description (Описание ссылки для установки флага)	Attend this event (участвовать в этом мероприятии)
Flagged message (Сообщение после установки флага)	You are attending this event (Вы согласились принять участие в этом мероприятии)
Unflag link text (Текст ссылки для снятия флага)	cancel attendance (отменить свое участие)
Unflag link description (Описание ссылки для снятия флага)	Cancel attendance to this event (Отменить свое участие в этом мероприятии)
Unflagged message (Сообщение после снятия флага)	You are no longer attending this event (Вы больше не участвуете в этом мероприятии)
What nodes this flag may be used on (Типы узлов, к которым может применяться этот флаг)	Event (Событие): отмечено Story (Статья): не отмечено
Display options (Настройки внешнего вида)	Display link on node teaser (Отображать ссылку в анонсе узла): не отмечено Display link on node page (Отображать ссылку на странице узла): отмечено Display checkbox on node edit form (Отображать флажок в форме редактирования): не отмечено

3. Щелчком на кнопке Submit (Отправить) создайте флаг attendance (участие). Теперь можно перейти к любому мероприятию, созданному ранее, и щелкнуть на ссылке attend this event (участвовать в этом мероприятии).

Практика: представление Attendee

Модуль Flag по умолчанию реализует представление, позволяющее любому пользователю получить список материалов, которые он отметил какими-либо предлагаемыми флагами. Теперь вы должны заметить ссылку My Attendance (Мои мероприятия) в навигационном меню, щелчок на которой откроет список всех мероприятий, которые были помечены вами для посещения.

Однако обществу книголюбов также необходимо иметь возможность получать списки всех желающих посетить то или иное мероприятие. Для этого мы создадим новое представление, изображенное на рис. 9.13.

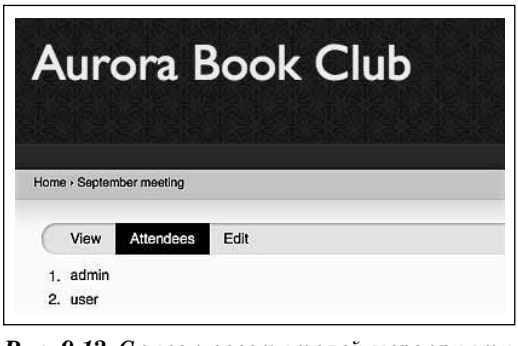


Рис. 9.13. *Список посетителей мероприятия*

1. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Views (*admin/build/views*) и щелкните на вкладке Add (Добавить).
2. Заполните форму в соответствии с табл. 9.18.

Таблица 9.18. Настройки представления Attendees (Участники)

Параметр	Значение
View name (Название представления)	attendees
View description (Описание представления)	Attendees for a given event (Участники мероприятия)
View tag (Рубрика представления)	event
View type (Тип представления)	User (Пользователь)

3. В первую очередь нам необходимо определить взаимосвязь, чтобы получить возможность извлекать информацию о флагах, установленных пользователями. Щелкните на ярлыке с изображением знака «плюс» в разделе Relationships (Взаимосвязи), выберите взаимосвязь Flags: User's flagged content (Флаги: Материалы, отмеченные пользователем) и щелкните на кнопке Add (Добавить). Заполните форму настройки взаимосвязи в соответствии с табл. 9.19.

Таблица 9.19. Настройки взаимосвязи Flags (Флаги)

Defaults: Relationship (По умолчанию: Взаимосвязи)	**Значение**
Flags: User (Флаги: Пользователь)	Label (Название): attendance Include only users who have flagged content (Включать только пользователей, отметивших материал): отмечено Flagged (Помечено флагом): Attendance (Участие)

4. Нам необходимо, чтобы представление отображало список пользователей, отметивших данный узел типа Event (Событие), поэтому мы должны добавить аргумент, в котором будет передаваться числовой идентификатор отмеченного узла. Для этого щелкните на ярлыке с изображением знака «плюс» в разделе Arguments (Аргументы). Выберите аргумент Flags: Content ID (Флаги: ID содержимого) и щелкните на кнопке Add (Добавить). Значения по умолчанию вполне подходят в нашем случае, поэтому можно просто щелкнуть на кнопке Update (Обновить).
5. Теперь можно добавить поля для отображения в списке. Щелкните на ярлыке с изображением знака «плюс» в разделе Fields (Поля). Выберите поле User: Name (Пользователь: Имя) и щелкните на кнопке Add (Добавить). Очистите поле Label (Название) и щелкните на кнопке Update (Обновить).
6. Было бы неплохо, если бы представление выводило нумерованный список участников, чтобы можно было быстро оценить общее число желающих принять участие в мероприятии. Для этого необходимо изменить стиль представления. Щелкните на раскрывающемся списке Style (Стиль) в разделе Basic settings (Базовые настройки), выберите пункт List (Список) и щелкните на кнопке Update (Обновить). Затем укажите тип списка Ordered list (Нумерованный список). Щелкните на кнопке Update (Обновить), чтобы сохранить изменения.
7. Чтобы завершить создание представления, необходимо добавить отображение Page (Страница). Выберите пункт Page (Страница) в раскрывающемся списке слева и щелкните на кнопке Add display (Добавить).

Заполните форму настройки отображения Page (Страница) в соответствии с табл. 9.20.

Таблица 9.20. Настройки отображения Page (Страница) для представления Attendees (Участники)

Параметр	Значение
Path (Путь)	node/%/attendees
Menu (Меню)	Type (Тип): Menu Tab (Вкладка меню) Title (Заголовок): Attendees (Участники)

8. Сохраните представление, настройки которого должны выглядеть, как показано на рис. 9.14.

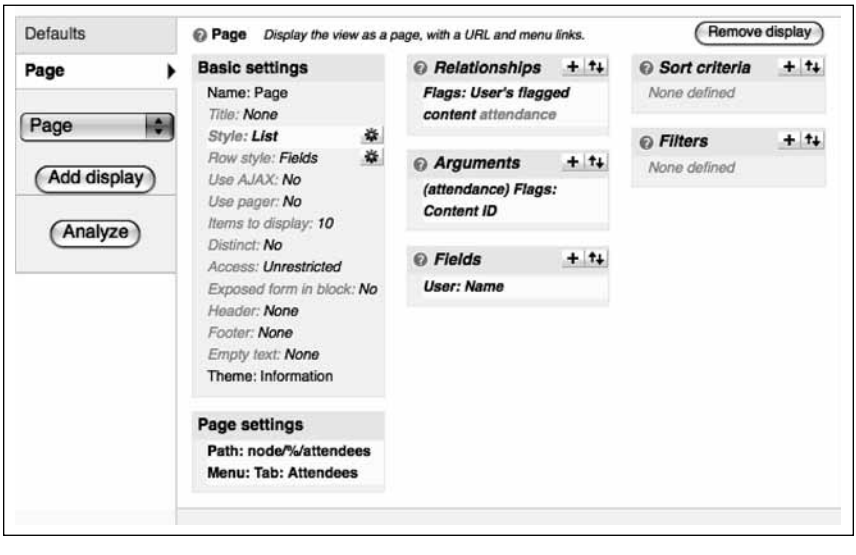


Рис. 9.14. *Настройки представления списка участников мероприятия*

Если теперь перейти на страницу с описанием мероприятия, можно увидеть вкладку Attendees (Участники). Щелчок на этой вкладке откроет список всех пользователей, заявивших о своем желании принять участие в мероприятии.

Дополнительно

Сайт, построенный нами, удовлетворяет всем требованиям общества. В процессе дальнейшей деятельности у членов общества могут появиться новые пожелания. Ниже приводится несколько модулей, которые могли бы помочь воплотить их будущие пожелания на сайте:

Calendar Popup (составная часть модуля Calendar)

Этот модуль также входит в состав пакета Calendar. Он позволяет реализовать вывод описания мероприятия после щелчка на ссылке в расписании не на отдельной странице, а во всплывающем окне JavaScript. Это удобно для тех, кто хотел бы подробнее ознакомиться с мероприятием, не покидая страницу расписания.

Countdown (http://drupal.org/project/countdown/)

Этот модуль добавляет блок, отображающий время, оставшееся до начала мероприятия. Это отличный способ дать людям знать, что следующая встреча состоится через четыре дня или через четыре часа.

Flag Actions (составная часть модуля Flag)

Модуль Flag можно настроить так, чтобы организовать рассылку извещений по электронной почте, а также удалять или снимать с публикации материалы по достижении определенного порогового количества установленных флагов. Эта особенность обычно используется для удаления спама или материалов с оскорбительным содержимым, однако она также может использоваться, чтобы, например, одного известить по электронной почте, что принять участие в мероприятии изъявили желание более 10 человек, а другого – что он должен захватить легкие закуски и напитки.

Location (http://drupal.org/project/location/) и GMap (http://drupal.org/project/gmap/)

Вместо того чтобы вводить адрес места проведения мероприятия в текстовое поле, можно задействовать модули Location и GMap, которые позволят использовать службу Google map для указания места проведения каждого мероприятия.

В заключение

В этой главе мы рассмотрели процесс создания сайта управления событиями для общества книголюбов Aurora Book Club, задействовав поле Date для пакета CCK, расширение Calendar для модуля Views и модуль Flag для получения информации об участниках. Общество книголюбов получило сайт, отображающий удобное расписание, а также возможность получать это расписание в формате iCal. Кроме того, сайт предлагает удобный доступ к списку всех участников отдельных мероприятий. Сайт получился весьма простым в использовании и прекрасно удовлетворяет все потребности членов общества.

Ниже приводится перечень модулей, упоминавшихся в этой главе:

- Calendar: *http://drupal.org/project/calendar*
- Countdown: *http://drupal.org/project/countdown*
- Date: *http://drupal.org/project/date*

- Event: *http://drupal.org/project/event*
- Flag: *http://drupal.org/project/flag*
- GMap: *http://drupal.org/project/gmap*
- Location: *http://drupal.org/project/location*
- Signup: *http://drupal.org/project/signup*
- Views: *http://drupal.org/project/views*

Дополнительные ресурсы:

- ADOdb Date Time Library: *http://phplens.com/phpeverywhere/adodb_date_library*
- Руководство для модуля Date: *http://drupal.org/node/92460*
- Модули, реализующие поддержку событий: *http://drupal.org/project/Modules/category/61*
- iCalendar: *http://en.wikipedia.org/wiki/ICalendar*
- Форматирование дат в языке PHP: *http://php.net/date*
- Функция `strtotime()` в языке PHP: *http://www.php.net/manual/en/function.strtotime.php*

10

Интернет-магазин

Многие фирмы, как крупные, так и небольшие, хотели бы извлекать максимум преимуществ от своего присутствия в Сети, предлагая приобрести свои товары и услуги прямо на сайте. Однако конструирование сайта электронной коммерции может оказаться довольно трудоемкой задачей. Существует несколько вариантов реализации подобных сайтов с разным уровнем сложности. Многие поставщики услуг хостинга предлагают свои пакеты электронной коммерции, стоимость которых может включаться в тарифный план или оплачиваться отдельно. Некоторые службы, такие как PayPal, предлагают простые способы оплаты покупок с использованием формы HTML, которая отправляется на обработку этим службам. Существуют и другие пакеты электронной коммерции, как свободно распространяемые, так и патентованные, которые вы можете установить и настроить. Самая большая проблема, связанная со всеми этими вариантами, заключается в отсутствии интеграции с остальной частью веб-сайта – все операции с покупательской корзиной и оплата производятся с привлечением внешних систем.

Еще одна сложность реализации сайта электронной коммерции заключается в привлечении реальных денег, как ваших, так и ваших покупателей. Покупатели могут предоставлять информацию о своих кредитных картах и другую секретную информацию, поэтому особое внимание следует уделить обеспечению безопасности.

В этой главе будут представлены следующие модули:

Ubercart (http://drupal.org/project/ubercart)

Полноценный пакет электронной коммерции для создания интернет-магазинов

Если вы собираетесь следовать за практическими упражнениями этой главы, вам следует установить Drupal, используя профиль установки Online Store, который установит сайт, создаст несколько пользовате-

лей и выполнит самые основные настройки. В окончательном виде веб-сайт будет выглядеть, как показано на рис. 10.1 и по адресу *http://store.usingdrupal.com*. За дополнительной информацией об использовании профилей установки обращайтесь к разделу «Вступление».

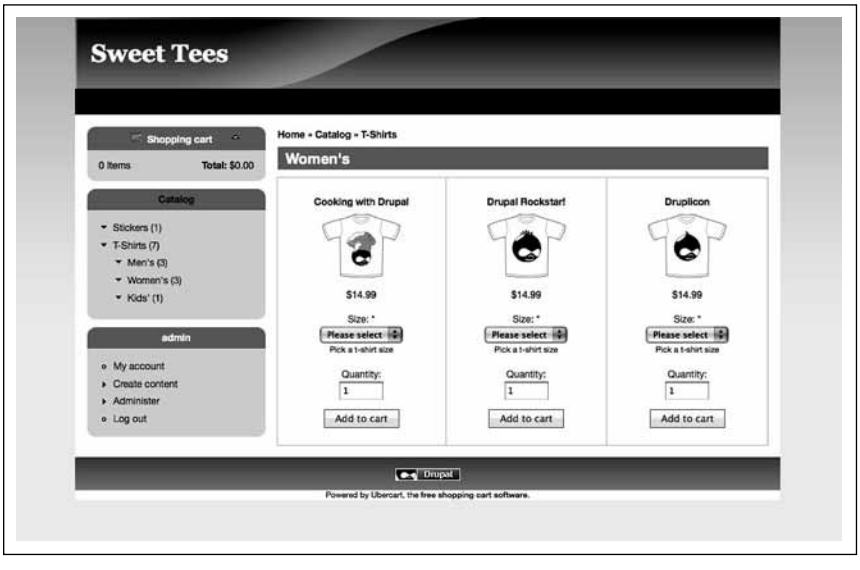


Рис. 10.1. *Окончательный вид веб-сайта Sweet Tees*

Задача

Sweet Tees – это местная фирма, занимающаяся продажей футболок, где можно купить весьма популярные футболки с необычными наклейками. Фирма имеет собственный магазин, и ее владельцы любят свое дело и свой город. Однако они часто получают заказы на футболки и наклейки по почте и хотели бы расширить свой бизнес в этом направлении. Прием заказов по телефону и отслеживание платежей оказалось хлопотным делом, отнимающим много времени как у владельцев фирмы, так и у покупателей.

Чтобы увеличить продажи, мы построим веб-сайт интернет-магазина Sweet Tees с поддержкой функциональности покупательской корзины, в которую посетители сайта смогут добавлять покупки, не будучи обязанными предварительно создать учетную запись. Покупательская корзина должна быть доступна на всех страницах в виде ссылки check out (купить). Функциональность магазина должна тесно интегрироваться с остальной частью веб-сайта, чтобы предоставить клиенту возможность никуда не переключаться в процессе покупки. Владельцы Sweet Tees хотели бы сделать процесс покупки максимально простым, поэтому от нас требуется реализовать одноступенчатый процесс покупки, не требующий от покупателя создавать учетную запись. Наконец,

владельцы хотели бы принимать оплату кредитными картами, поэтому нам необходимо будет реализовать процедуру оплаты.

Примечания к реализации

У фирмы Sweet Tees имеется несколько вариантов организации управления своим интернет-магазином, но в общем владельцы хотели бы предоставить своим клиентам простой и дружественный интерфейс. Им требуется решение, которое было бы простым, элегантным и, вместе с тем, комплексным.

Для системы Drupal существует два базовых решения задачи электронной коммерции, позволяющих реализовать требуемые особенности, каждое из которых состоит из нескольких модулей.

Модуль e-Commerce (*http://drupal.org/project/ecommerce*) представляет собой более старое и при этом более гибкое решение. Этот пакет создан в виде модульной платформы для построения решений электронной коммерции в системе Drupal. В некотором смысле, это отдельная платформа разработки, построенная поверх платформы Drupal. Основное преимущество пакета e-Commerce заключается в непревзойденной управляемости и расширяемости. Интернет-магазин включает в себя множество разнообразных компонентов (система обработки платежей, вычисление стоимости доставки, налогов и так далее), поэтому такая высокая гибкость оказывается очень кстати. Однако высокая гибкость повышает уровень сложности, в результате чего установка и настройка пакета занимают много времени. Кроме того, из-за архитектурных решений процедура покупки в пакете e-Commerce реализована в виде последовательности из нескольких страниц и требует, чтобы покупатель создал свою учетную запись на веб-сайте. Это нарушает два основных требования, предъявляемых заказчиками.

Решение Ubercart (*http://drupal.org/project/ubercart*) гораздо более современное и изначально проектировалось с целью обеспечить большую простоту установки, настройки и сопровождения по сравнению с пакетом e-Commerce. Несмотря на меньшую гибкость, пакет Ubercart способен удовлетворить потребности 80–90% интернет-магазинов. Кроме того, пакет Ubercart обладает рядом замечательных особенностей, которые делают его особенно привлекательным для нас: одноступенчатая процедура оформления покупки, возможность приобретения товаров анонимными пользователями и превосходный административный интерфейс.

Чтобы создать простое и изящное представительство магазина Sweet Tees в Интернете, мы будем использовать пакет Ubercart.

Проект Ubercart имеет собственный веб-сайт, отдельный от сайта Drupal.org, где вы найдете форум поддержки, документацию и дополнительные модули. Посетите его по адресу *http://www.ubercart.org*.

В центре внимания: пакет Ubercart

Ubercart – это полный комплект модулей для создания интернет-магазина. Он содержит более 30 отдельных модулей, реализующих все особенности, необходимые для конструирования интернет-магазина. В состав пакета входят как самые основные модули, такие как Payment или Cart, так и узкоспециализированные, реализующие специфические функции. Исследование гигантского списка модулей (число которых превышает число модулей, составляющих ядро Drupal!) может оказаться весьма трудоемким делом, поэтому в данном разделе для каждого рассматриваемого модуля мы обозначим его назначение и где он применяется.

Модули, составляющие пакет Ubercart, разбиты на пять категорий и представлены на странице Administer (Администрирование)→Site building (Конструкция сайта)→Modules (Модули) (*admin/build/modules*). Здесь мы расскажем, какие модули входят в каждую категорию.

Ubercart – core

Категория Ubercart – core (Ubercart – ядро) содержит базовые компоненты платформы для построения интернет-магазина. В эту категорию включено пять модулей: Cart, Conditional Actions, Order, Product и Store. Для создания интернет-магазина необходимы все эти модули. Ниже приводится краткое описание каждого из них:

- Модуль Cart реализует функциональность покупательской корзины, внешний вид которой приводится на рис. 10.2, и обладает такими особенностями, как отслеживание товаров, выбранных покупателем, и количество заказанных экземпляров каждого продукта. Кроме того, модуль Cart отвечает за реализацию процедуры «покупки». Помимо базовой функциональности, модуль Cart обеспечивает возможность передачи функций покупательской корзины и покупки другим модулям.

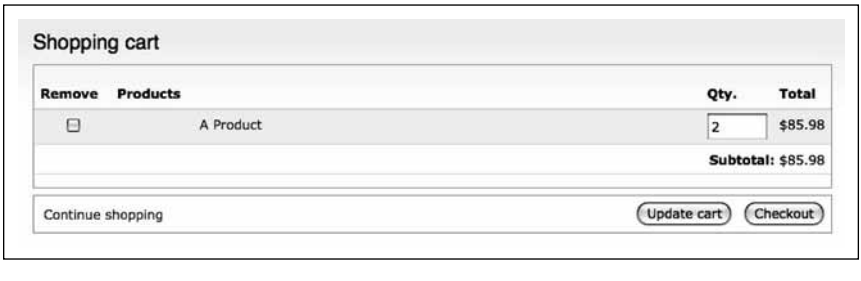


Рис. 10.2. *Покупательская корзина, реализованная модулем Cart*

- Модуль Conditional Actions реализует механизм выполнения операций для остальной части пакета Ubercart, позволяя определять на-

страиваемые действия, основанные на правилах, такие как вычисление суммы налога и стоимости доставки. Этот модуль опирается на систему действий ядра Drupal и является улучшенной альтернативой модулю Trigger, о котором рассказывалось в главе 6.

- Модуль Order отвечает за хранение, отслеживание и управление отдельными заказами. На рис. 10.3 показана форма заказа, созданная модулем Order. Модуль Order обеспечивает возможность создания заказа (когда заказ принимается по телефону) и выставления счета вручную, а также предоставляет интерфейс для просмотра и изменения существующих заказов. Как и модуль Cart, модуль Order также служит базовым модулем для остальной части пакета, предоставляя функции для автоматизированного выполнения и обработки платежей.

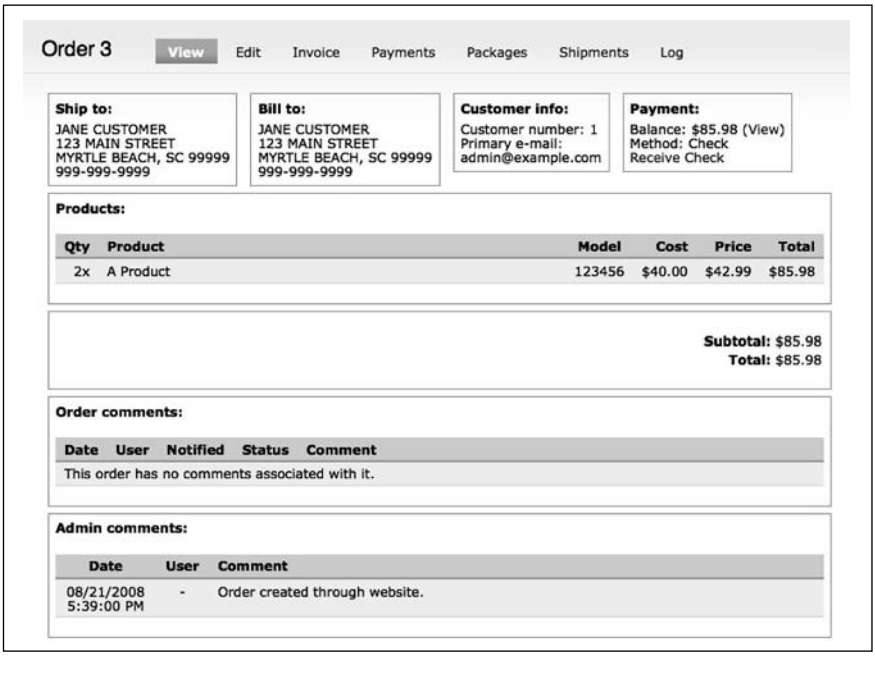


Рис. 10.3. *Форма заказа, созданная модулем Order*

- Модуль Product обеспечивает хранение информации обо всех товарах, которые можно приобрести в интернет-магазине. Он создает тип материала «продукт», который может расширяться за счет определения «классов продуктов». На рис. 10.4 показаны некоторые характеристики о продукте, предоставляемые модулем Product.
- Модуль Store реализует административный интерфейс управления всем пакетом Ubercart. Он создает простой интерфейс управления по адресу Administer (Администрирование)→Store administration

(Управление магазином) (*admin/store*), изображенный на рис. 10.5, а также предоставляет общие вспомогательные функции, такие как преобразование единиц измерения и учет других особенностей, отличающихся от страны к стране.

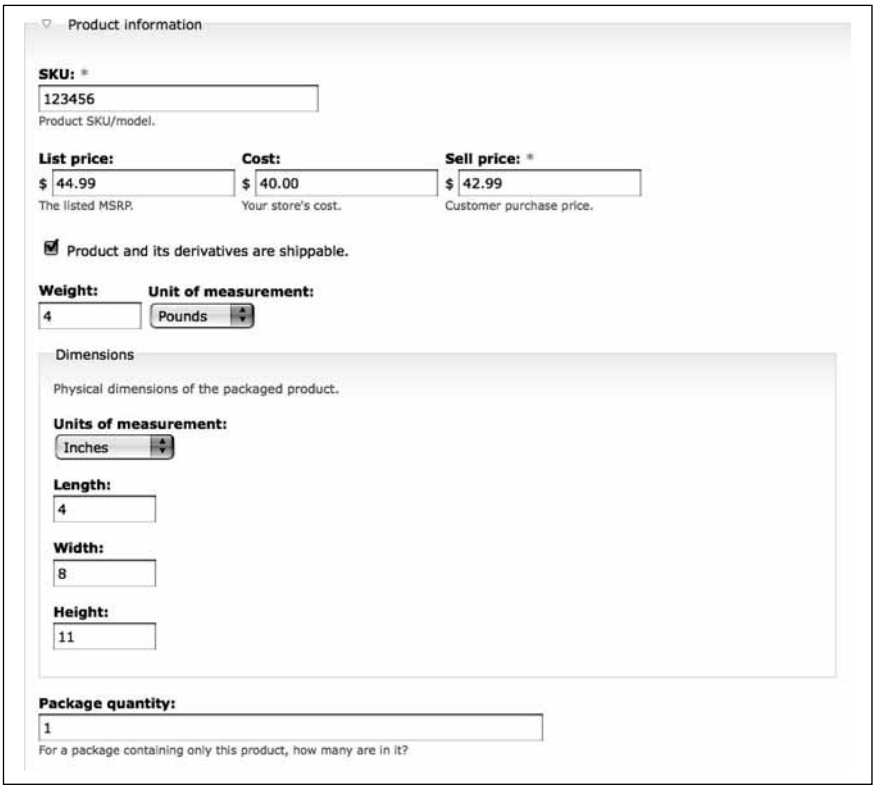


Рис. 10.4. *Информация о продукте, предоставляемая модулем Product*

Ubercart – core (optional)

Категория Ubercart – core (optional) (Ubercart – ядро (дополнительные)) содержит модули, большая часть которых, но не все, необходима для создания интернет-магазина. В этом разделе мы опишем некоторые интересные и наиболее востребованные особенности, предоставляемые этими модулями.

- Модуль Attribute позволяет присваивать продуктам различные характеристики, позволяющие покупателям указывать желаемые параметры, вместо того чтобы заносить в список отдельные продукты, имеющие незначительные отличия. Ниже мы будем использовать модуль Attributes, чтобы дать покупателям возможность при заказе футболок выбирать размер.

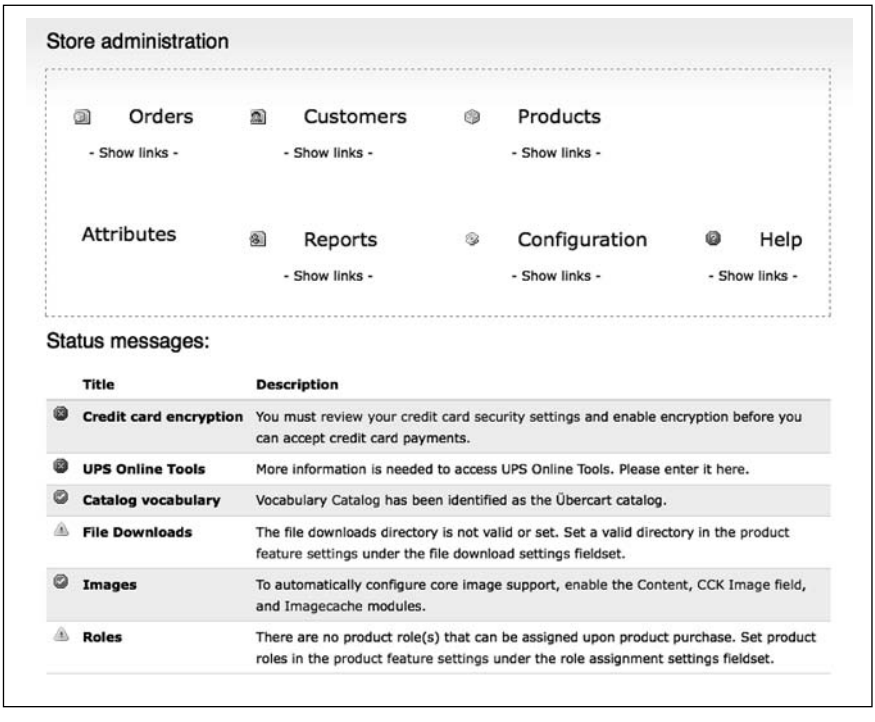


Рис. 10.5. *Панель администрирования, предоставляемая модулем Store*

- Модуль Catalog обеспечивает возможность классификации продуктов с помощью механизма таксономии Drupal, посредством создания предопределенного словаря «Product Categories» (Категории продуктов). Кроме того, модуль Catalog реализует некоторые дополнительные особенности, такие как отображение изображений, связанных с каждой из категорий (как показано на рис. 10.6), и некоторые дополнительные возможности просмотра характеристик продуктов в каталоге.

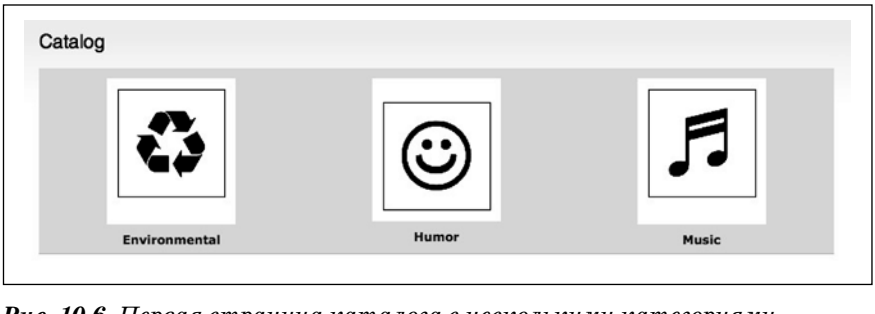


Рис. 10.6. *Первая страница каталога с несколькими категориями*

- Модуль File Downloads позволяет прилагать к продуктам загружаемые файлы. Эта возможность идеально подходит для продажи программного обеспечения или цифровых материалов. Модуль File Downloads автоматически отправляет покупателю по электронной почте ссылку для безопасной загрузки файла. Кроме того, покупатели могут повторно загрузить любые приобретенные продукты, зарегистрировавшись под своей учетной записью.
- Модуль Notify обеспечивает возможность отправки по электронной почте извещений о произведенной покупке или об изменении заказа.
- Модуль Payment реализует платформу для организации оплаты покупок, обеспечивая интеграцию Ubercart со сторонними службами электронных платежей. При выполнении покупки этот модуль позволяет покупателю выбрать наиболее удобный способ оплаты из списка доступных, как показано на рис. 10.7. Кроме того, модуль Payment обеспечивает возможность тестирования, что очень удобно при отладке процедуры покупки (без привлечения сторонних служб). Все модули электронных платежей, входящие в состав категории Ubercart – payment (Ubercart – платежи), тесно интегрируются с базовым модулем Payment. Большая часть этих модулей обеспечивает интеграцию с определенными службами электронных платежей, такими как 2Checkout, Authorize.net, CyberSource и PayPal, с некоторыми исключениями:
  - Модуль Test Gateway обеспечивает простой интерфейс к фиктивной системе обработки платежей, который удобно использовать для отладки магазина, чтобы убедиться, что заказы и отчеты работают должным образом. Модуль Test Gateway не выполняет фактических операций с кредитной картой.

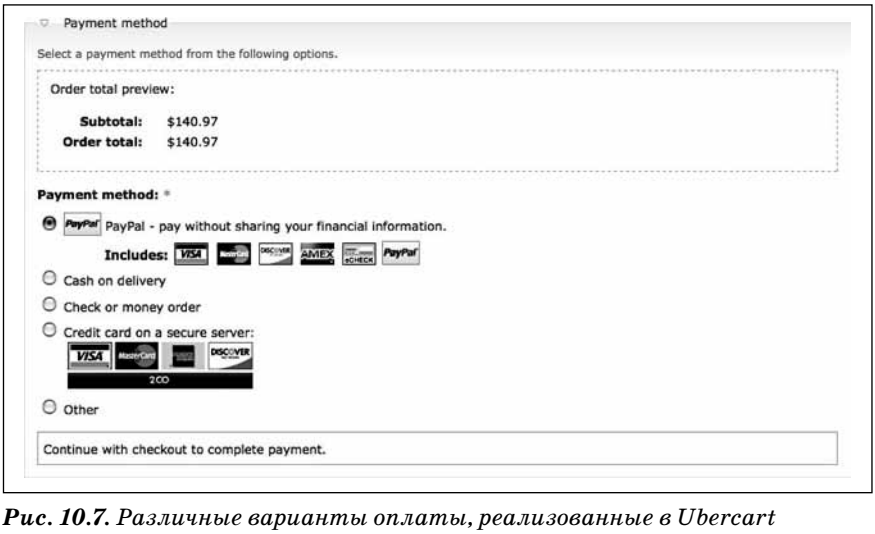


Рис. 10.7. *Различные варианты оплаты, реализованные в Ubercart*

  - Модуль Payment Method Pack добавляет «Другие» варианты оплаты, такие как COD (наложенный платеж), оплата наличными или денежным переводом. Этот модуль не выполняет перевод денег напрямую.
  - Модуль Recurring Payments сам не взимает платежи, но может добавлять к некоторым продуктам суммы периодических платежей, что позволяет на базе пакета Ubercart реализовать предоставление платных услуг по подписке.
  - Модуль Credit Card – это базовый модуль, добавляющий в процедуру покупки возможность «оплаты по кредитной карте». Он предоставляет средства интеграции со специфическими сторонними модулями электронных платежей. Сам по себе модуль Credit Card не выполняет никаких операций с кредитными картами.
- Модуль Reports предоставляет консоль отчетов для составления отчетов о продажах и общей деятельности электронного магазина, как показано на рис. 10.8. Кроме того, этот модуль позволяет загружать отчеты в формате CSV (Comma-Separated Values – значения, разделенные запятыми) для последующего импортирования в приложения электронных таблиц и другие инструменты работы с отчетами.

Order statuses used: Completed

#	Product	Views	Sold	Revenue	Gross
1	A Product	2	**2**	**$85.98**	**$5.98**
	123456		2	$85.98	$5.98
2	Another product	1	**0**	**$0.00**	**$0.00**
	987654		0	$0.00	$0.00
3	Yet another product	3	**0**	**$0.00**	**$0.00**
	x		0	$0.00	$0.00

Export to CSV file. Show all records

Рис. 10.8. *Пример отчета, сгенерированного модулем Reports*

- Модуль Roles дает возможность присваивать пользователям, приобретающим определенные продукты, указанные роли на неопределенный или ограниченный срок. Этот модуль отлично подходит для реализации поддержки членства или предоставления услуг по подписке.
- Модуль Shipping во многом напоминает модуль Payment – он предоставляет прикладной программный интерфейс, или основу управления подготовкой приобретенных продуктов к доставке. Все модули, перечисленные в категории Ubercart – fulfillment (Ubercart – исполнение), интегрируются с модулями Shipping и/или Shipping Quotes. Их назначение состоит в том, чтобы обеспечить интеграцию со специфическими модулями третьих служб, выполняющими вычисление стоимости доставки, генерирование маркировочных наклеек или созда-

ние номеров отслеживания. В состав пакета Ubercart включены модули интеграции с двумя основными службами доставки: U.S. Postal Service (почтовая служба США), посредством ее веб-службы XML, и UPS (United Parcel Service – объединенная посылочная служба), посредством аналогичной службы. Имеются также дополнительные модули, позволяющие оценивать стоимость доставки на основе фиксированного тарифа (умножаемого на количество продуктов) и оплаты посылки по весу, с учетом веса каждого продукта. На рис. 10.9 показаны некоторые варианты оформления доставки, предоставляемые пакетом Ubercart.

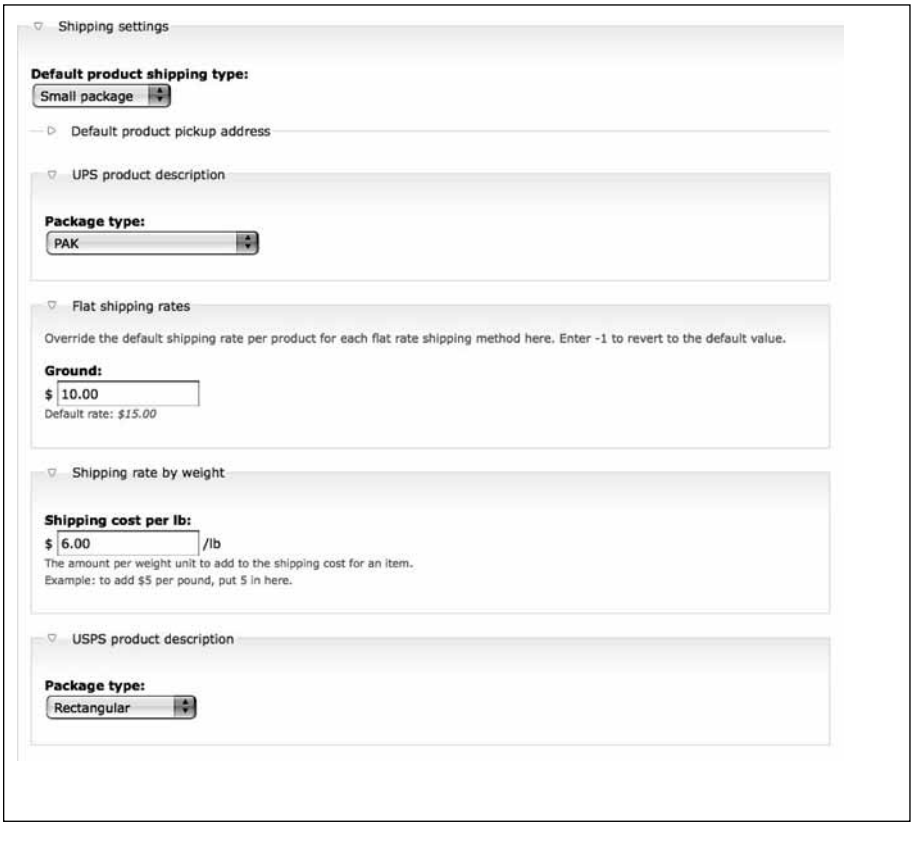


Рис. 10.9. *Варианты доставки продукта*

Дополнительные модули интеграции со службами доставки можно загрузить на сайте Ubercart, в разделе Contributions (Сторонние модули), по адресу *http://www.ubercart.org/contrib*.

- Модуль Shipping Quotes также предоставляет базовые средства для оценивания стоимости доставки. Другие модули исполнения заказов,

перечисленные в категории Ubercart – fulfillment (Ubercart – исполнение), могут предоставлять свои собственные, уникальные методы вычислений, исходя из метода доставки, выбранного покупателем.

- Модуль Taxes реализует интерфейс для создания правил исчисления налогов для определенных регионов, исходя из типов продуктов.

Ubercart – extra

Большая часть модулей из категории Ubercart – extra (Ubercart – дополнительные) не будет задействована на нашем сайте, тем не менее среди них имеются весьма интересные модули. Грань между категориями «extra» (дополнительные) и «core (optional)» (ядро (дополнительные)) весьма расплывчата. Однако в целом модули из категории «core (optional)» (ядро (дополнительные)) непосредственно участвуют в организации процесса покупки или реализуют дополнительные расширяемые элементы платформы. А модули из категории «extra» (дополнительные) обеспечивают дополнительные функциональные возможности:

- Модуль Cart Links позволяет администраторам конструировать адреса URL для ссылок, которые будут добавлять определенное количество указанных продуктов (наряду с дополнительной информацией). Затем эти ссылки можно будет использовать для размещения на внешних сайтах, в блогах, на сайтах филиалов или на других страницах сайта интернет-магазина, чтобы предоставить посетителям ссылки, действующие по принципу «купить прямо сейчас». Для генерации таких ссылок не предусматривается пользовательский интерфейс, но вы можете поближе познакомиться с модулем Cart Links на странице Administer (Администрирование)→Store administration (Управление магазином)→Help (Справка)→Creating cart links (Создание ссылок для покупки) (*admin/store/help/cart_links*), после включения модуля.
- Модуль Google Analytics for Ubercart обеспечивает интеграцию Ubercart со службой Google Analytics и может использоваться для отслеживания курсов валют и уровней продаж торгующих компаний. Этот модуль требует установки модуля Google Analytics (*http://drupal.org/project/google_analytics*).
- Модуль Importer – это очень удобный модуль, обеспечивающий возможность обмена данными между Ubercart и внешними системами. Модуль Importer фактически реализует функции импорта и экспорта информации о продуктах в формате XML.
- Модуль Product Kit обеспечивает возможность объединения продуктов в один пакет, который продается как единица товара. Для таких пакетов может быть установлена специальная цена (со скидкой).
- Модуль Repeater предоставляет возможность обновлять информацию о продукте из удаленных интернет-магазинов, созданных на основе пакета Ubercart. Этот модуль очень удобно использовать, ког-

да пакет Ubercart работает в сети электронных магазинов, торгующих похожими или сопутствующими товарами. Обновление информации о продуктах производится с применением того же формата, который использует модуль Importer.

- Модуль Stock обеспечивает инструменты, позволяющие следить и управлять уровнем запасов на складе. Администраторы интернет-магазина могут определять нижние пороговые значения запасов продуктов, по достижении которых они будут извещаться по электронной почте о слишком низких запасах на складе. Модуль Stock также интегрируется с модулем Reports, что позволяет получать отчеты о запасах товаров.

В центре внимания: панель администрирования Ubercart

Пакет Ubercart, в значительной степени из-за его сложности и значительного числа модулей, имеет собственный административный раздел, который находится на странице Administer (Администрирование)→Store administration (Управление магазином) (*admin/store*), изображенной на рис. 10.10. Управление всеми особенностями Ubercart, о которых будет рассказываться ниже, осуществляется в этом новом разделе.

В главной панели отображается список всех доступных операций, которые могут выполняться для управления магазином, включая управление каталогом товаров, просмотр и заполнение заказов, просмотр отчетов о деятельности магазина и корректировку любых параметров настройки. Разделы с дополнительными ссылками по умолчанию свернуты, их можно развернуть щелчком на ссылках Show links (Показать ссылки). Ниже главной панели присутствует список сообщений о состоянии, которые могут указывать на наличие каких-либо проблем или на невыполнение некоторых этапов настройки. Эти сообщения могут помочь разобраться, когда что-то работает не так, как ожидалось.

Ссылки Orders (Заказы), Customers (Клиенты) и Products (Товары) указывают на раздел, где можно создавать записи, управлять существующими записями или выполнять поиск по ним. Это место, где выполняются ежедневные работы по администрированию.

Раздел Reports (Отчеты) позволяет получить ответы на вопросы о положении дел в магазине. Какие товары продаются лучше всего? Достаточно ли товаров на складе или пора заказать еще? Каков объем прибыли? Ответы на эти и другие вопросы можно получить с помощью инструментов, доступных в этом разделе.

Раздел Configuration (Настройка) содержит огромное количество настроек всего, начиная от настройки переадресации пользователей, после того как они добавят какой-либо товар в корзину, до настройки внешне-

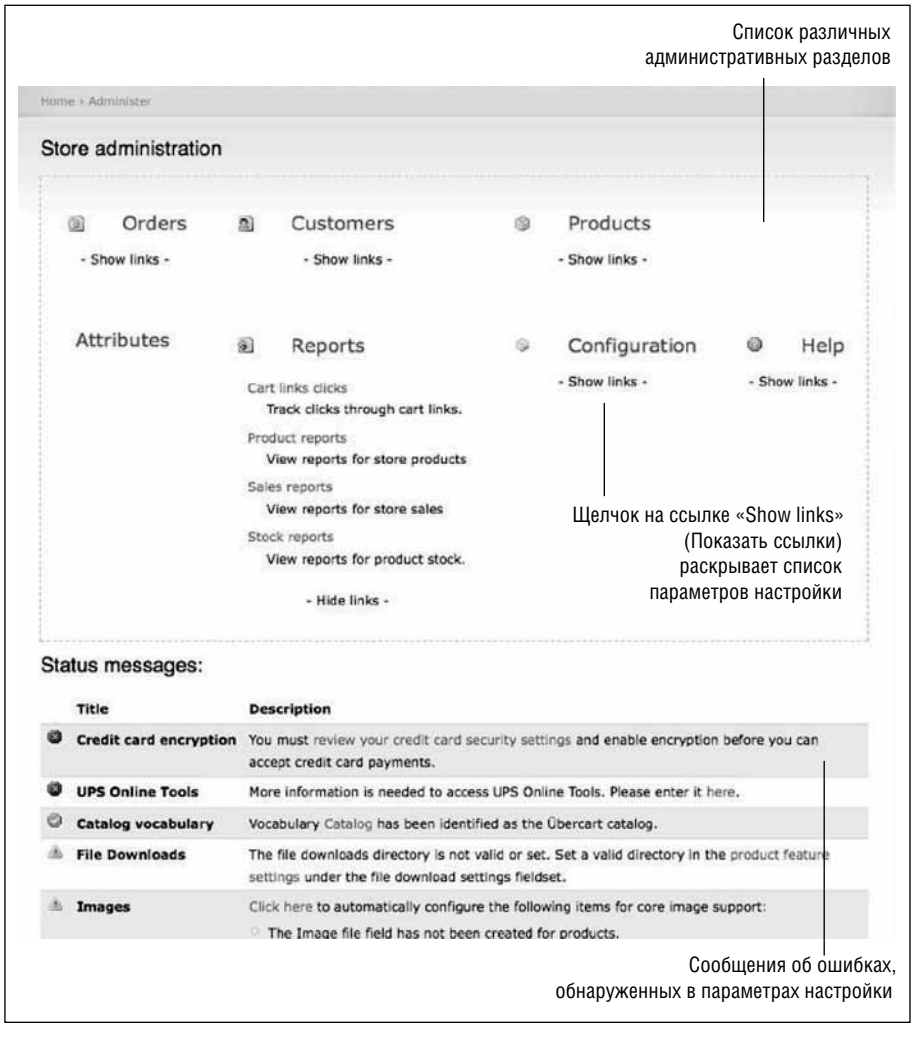


Рис. 10.10. *Панель администрирования пакета Ubercart*

го вида формы ввода адреса доставки и количества заказов, отображаемых одновременно в форме просмотра заказов. Если вам потребуется изменить какой-либо аспект Ubercart, вы наверняка найдете здесь соответствующие настройки. В отличие от большинства административных разделов Drupal, раздел Store settings (Настройки магазина) содержит список всех установленных значений параметров, сгруппированных в категории. Чтобы изменить какое-либо значение, достаточно просто щелкнуть на названии категории (которое выделяется при наведении указателя мыши), как показано на рис. 10.11.

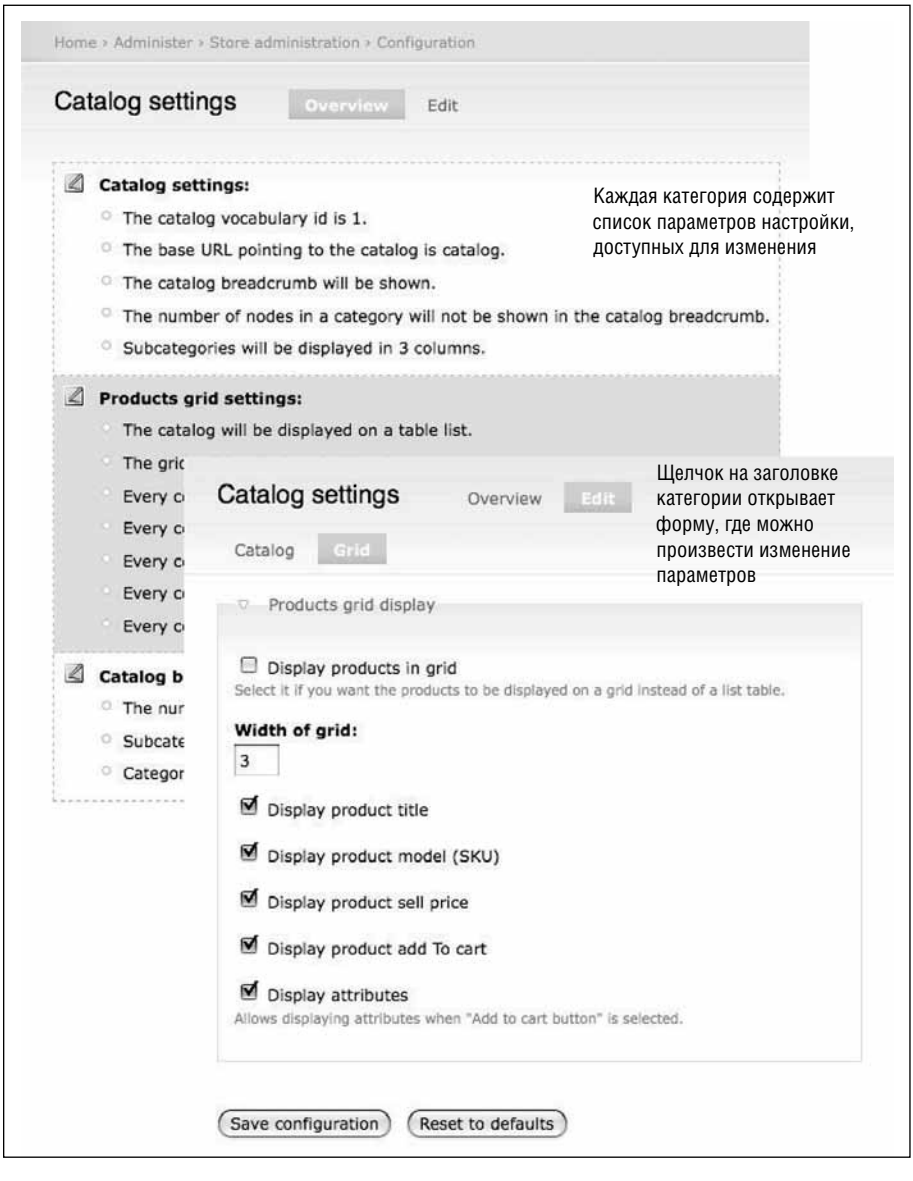


Рис. 10.11. *Страницы настройки пакета Ubercart отображают списки категорий параметров, щелчок на которых открывает соответствующие формы редактирования*

И наконец, раздел Help (Справка) содержит список ресурсов Ubercart, а также ссылки на страницы справки для конкретных модулей, входящих в состав пакета Ubercart: сюда стоит обращаться в первую очередь,

столкнувшись с проблемами, если вам необходима помощь от других пользователей или экспертов Ubercart.

Практика: настройка магазина

Прежде чем открыть интернет-магазин по продаже футболок, необходимо сначала указать некоторую базовую информацию, которая в дальнейшем будет использоваться другими особенностями. Начнем с самых основ: с информации, которая будет использоваться в счетах и для вычисления стоимости доставки.

Начальная настройка

1. Зарегистрируйтесь на сайте Sweet Tees как пользователь *admin*, если вы до сих пор этого не сделали.
2. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Modules (Модули) (*admin/build/modules*) и включите следующий модуль:
   - Ubercart – core (Ubercart – ядро)
     - Store
3. Перейдите на страницу Administer (Администрирование)→Store administration (Управление магазином)→Configuration (Настройка)→Store settings (Настройки магазина) (*admin/store/settings/store*).
4. Щелкните на заголовке категории Name and contact information (Название и контактная информация), в результате чего откроется форма Contact settings (Настройки контактов) (*admin/store/settings/store/edit*). Заполните форму в соответствии с табл. 10.1 и по завершении щелкните на кнопке Save configuration (Сохранить конфигурацию). Когда вы вернетесь на страницу Overview (Обзор) в разделе Store settings (Настройки магазина), она должна выглядеть, как показано на рис. 10.12.

Таблица 10.1. Настройки контактов

Параметр	Значение
Store name (Название магазина)	Sweet Tees
Store owner (Владелец)	Stephen Sweet
E-mail address (Адрес эл. почты)	store@example.com
Phone number (Телефон)	800-555-1234
Fax number (Факс)	110-555-4321
Street address 1 (Адрес 1)	123 Example St
Street address 2 (Адрес 2)	

City (Город)	Example City
Country (Страна)	United States
Zone (Область, край)	California
Postal code (Почтовый индекс)	90210

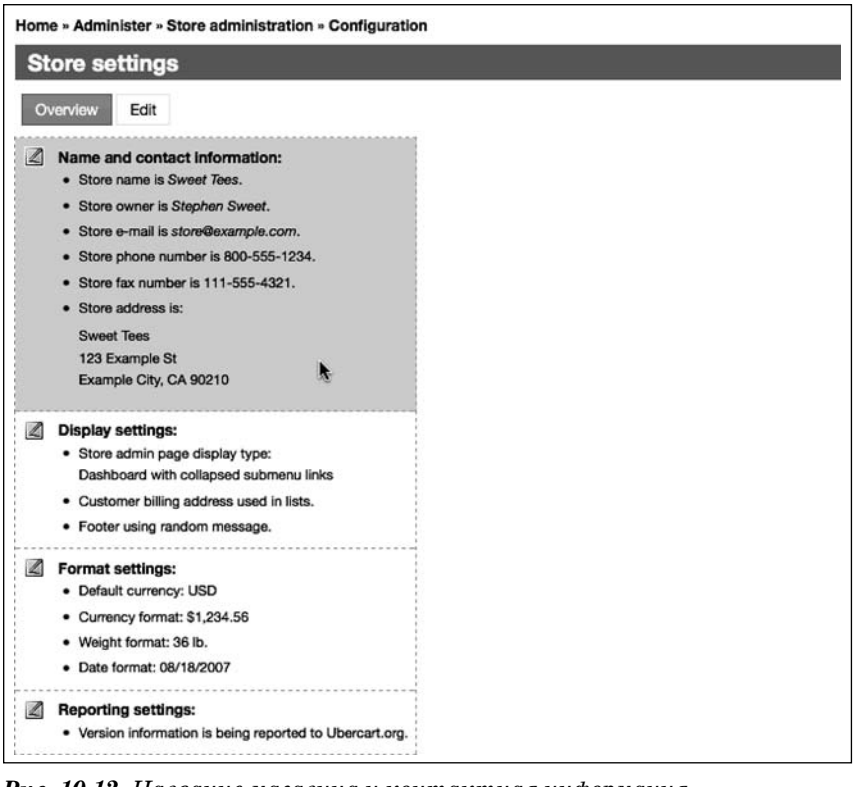


Рис. 10.12. *Название магазина и контактная информация*

Несмотря на то что по умолчанию пакет Ubercart обеспечивает поддержку значений параметров, принятых в Северной Америке, тем не менее они допускают возможность полной настройки под особенности любой другой страны. На вкладке Format settings (Настройка форматов) (*admin/store/settings/store/edit/format*) можно настроить отображение различных единиц измерения, таких как валюта, вес, длина и дата. А в категории Country settings (Региональные настройки), на странице Administer (Администрирование)→Store administration (Управление магазином)→Country settings (Региональные настройки) (*admin/store/settings/countries*), можно импортировать дополнительные региональные данные и установить формы ввода контактной информации для конкретной страны.

В центре внимания: модули Products, Product Classes и Attributes

Прежде чем приступить к добавлению товаров в каталог магазина, необходимо потратить некоторое время на изучение вопроса, как система Ubercart интерпретирует товары. Товары в Ubercart – это узлы, то есть с товарами можно делать все, что мы делали с узлами в этой книге: можно добавлять к товарам комментарии или оценки, маркировать их, добавлять поля CCK для хранения дополнительных свойств, выводить списки товаров с помощью модуля Views и так далее. Такая гладкая интеграция товаров магазина с остальным содержимым сайта, которым может управлять система Drupal, является самой «убойной» особенностью пакета Ubercart.

Модуль Product определяет единственный тип узла «Product» (Продукт), в состав которого уже входят такие поля, как SKU (артикул, или код товара) и цена, как показано на рис. 10.13. Эти поля напрямую взаимодействуют с остальными частями системы, такими как отчеты и расчет стоимости доставки.

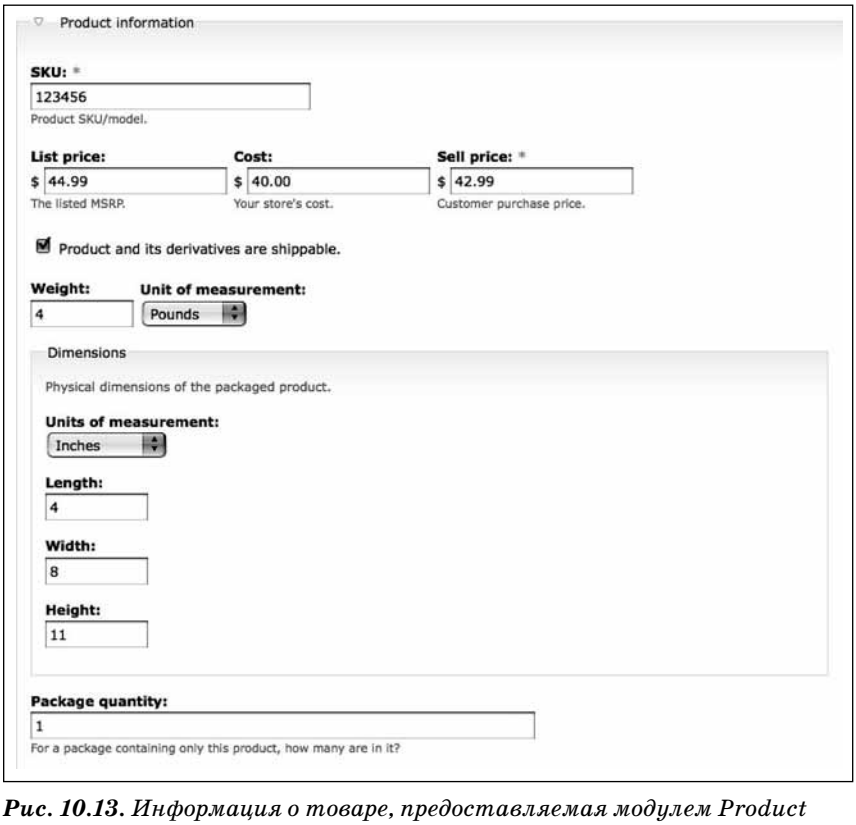


Рис. 10.13. *Информация о товаре, предоставляемая модулем Product*

Такого типа узла будет вполне достаточно, если магазин продает однотипные товары, такие как членство в клубе. Однако многие интернет-магазины имеют гораздо более сложную организацию. Например, Amazon.com продает книги, фильмы и, как мы видели в главе 4, товары для кухни. Книги обладают такими свойствами, как «автор» и «номер ISBN». Фильмы могут иметь такие свойства, как «оценка» и «киностудия». Представьте себе, какой длины была бы форма для ввода информации, если бы потребовалось предоставить поля для всех этих свойств в каждом товаре, для всех возможных типов товаров. Нет уж, спасибо.

К счастью, разработчики Ubercart нашли выход из этого затруднительного положения: специальные типы узлов, которые называются *классы продуктов*. Классы продуктов несколько отличаются от стандартных типов содержимого, так как для них необходимо наследовать поля базового продукта. Однако после создания класса продукта он выглядит и ведет себя, как обычный тип содержимого в Drupal – для него создается форма ввода на странице Create content (Создать материал) (*node/add*), и появляется возможность изменять поля и свойства на странице Administer (Администрирование)→Content management (Содержание)→Content types (Типы материалов) (*admin/content/types*). Каждый класс продуктов может настраиваться, не оказывая влияния на другие продукты. Вы можете создать классы продуктов «Книга» и «Фильм» и задействовать модуль CCK, чтобы придать каждому из них свои, характерные свойства. Мы будем использовать классы продуктов для создания наших продуктов «футболка» и «наклейка».

Чтобы пакет Ubercart мог опознать тип узла, как продукт, это должен быть либо тип узла «Product» (Продукт), определяемый модулем Product, либо «класс продуктов». Без этого для различных свойств продуктов, таких как артикул или цена, не будут отображаться поля ввода.

А как быть с такими товарами, как футболки, которые фактически являются однотипными товарами, но могут отличаться по цвету и размеру? Эти отличия, хотя и важные для исполнения заказа и, иногда, цены, в действительности определяют не отдельные типы товаров, а лишь различные разновидности одного и того же товара. Безусловно, было бы слишком утомительно создавать отдельные продукты, такие как «Красная футболка с маленьким логотипом Drupal», «Красная футболка с большим логотипом Drupal», «Белая футболка с маленьким логотипом Drupal» и так далее.

Все эти незначительные отличия в терминологии Ubercart называются *атрибутами*, которые реализуются модулем Attribute. Для каждого атрибута, такого как «Цвет», определяется набор возможных значений, таких как «Красный», «Синий» и «В клетку», которые может выбирать покупатель при добавлении товара в свою покупательскую корзину. Атрибуты могут совместно использоваться различными класса-

ми продуктов (цвет могут иметь и наклейки, и футболки) или быть характерными только для одного типа продуктов. Вы можете даже устанавливать разные цены для различных значений атрибута, так как, например, футболки в клетку могут шиться вручную. На рис. 10.14 приводится пример атрибута «Media format» (Формат записи), который может применяться к альбомам. Компакт-диск – это вполне реальный предмет, имеющий себестоимость и вес. Напротив, MP3 – это цифровой формат, который не имеет этих свойств.

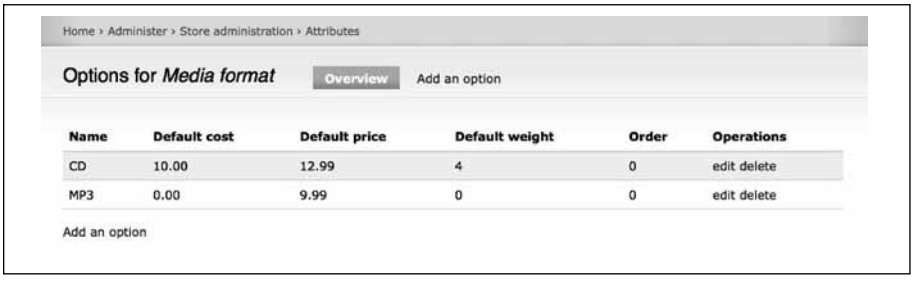


Рис. 10.14. *Пример атрибута с различными вариантами значений*

Практика: создание продуктов

В этом разделе мы определим информацию о продукте для сайта Sweet Tees, что является первым обязательным шагом на пути к созданию интернет-магазина. Модуль Product из пакета Ubercart реализует тип содержимого Product (Продукт), а модули FileField (*http://drupal.org/project/filefield*), ImageField (*http://drupal.org/project/imagefield*) и ImageCache (*http://drupal.org/project/imagecache*), о которых рассказывалось в главе 7, позволят нам отобразить коллекцию изображений продуктов.

Пакет Ubercart использует модули ImageField и ImageCache для работы с изображениями продуктов, поэтому мы можем изменять любые настройки по умолчанию, используя приемы, описанные в главе 7, чтобы приспособить их под свои цели.

Начальная настройка

Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Modules (Модули) (*admin/build/modules*) и включите следующие модули:

- CCK
  - Content
  - FileField
  - ImageField

- Core (Ядро)
  - Path
- ImageCache
  - ImageAPI
  - ImageAPI GD2
  - ImageCache
  - ImageCache UI
- Other (Другой)
  - Token
- Ubercart – core (Ubercart – ядро)
  - Product
- Ubercart – core (optional) (Ubercart – ядро (дополнительные))
  - Attribute
  - Catalog

После включения этих модулей можно заметить, насколько существенно изменился раздел администрирования магазина. Теперь на странице Administer (Администрирование)→Store administration (Управление магазином) (*admin/store*) в разделе Status messages (Сообщения о состоянии) отображаются сообщения и появились две новые категории, Products (Продукты) и Attributes (Атрибуты), как показано на рис. 10.15.

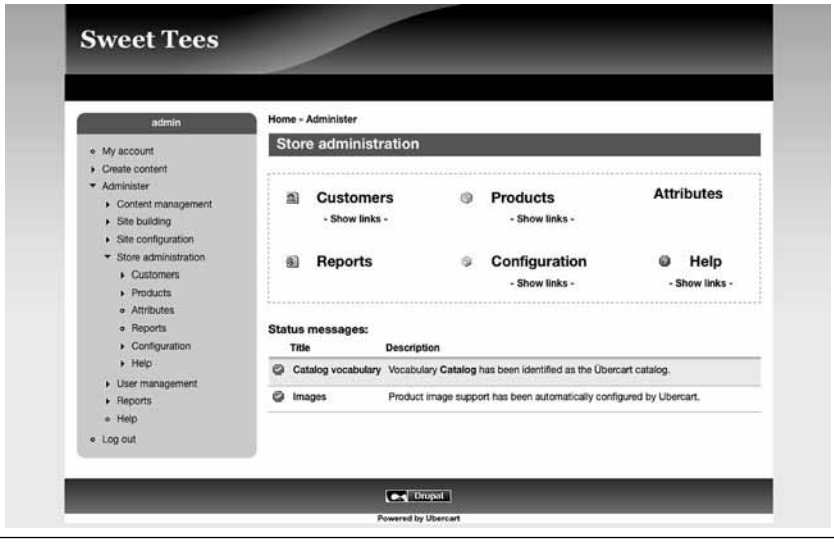


Рис. 10.15. *Панель администрирования магазина после включения модуля Product и сопутствующих ему модулей*

Настройка классов продуктов

Когда на предыдущем шаге мы включили модули, пакет Ubercart автоматически добавил тип содержимого Product (Продукт) и связал его с полем Image (Изображение), чтобы обеспечить возможность присоединения изображений к продуктам. Однако фирма Sweet Tees продает два типа продуктов: футболки и наклейки. Чтобы учесть это обстоятельство, мы создадим два класса продуктов, как показано на рис. 10.16.

Рис. 10.16. *Классы продуктов для футболок и наклеек*

1. Перейдите на страницу Administer (Администрирование)→Store administration (Управление магазином)→Products (Продукты)→Manage classes (Управление классами) (*admin/store/products/classes*).
2. Перед вами откроется форма добавления класса. Создайте первый класс для футболок, в соответствии с табл. 10.2, и щелкните на кнопке Submit (Отправить).

Таблица 10.2. Добавление класса футболок

Параметр	**Значение**
Class ID (Идентификатор класса)	tshirt
Class Name (Название класса)	T-shirt (Футболка)
Description (Описание)	Our award-winning T-shirts (Наши замечательные футболки)

3. Теперь добавьте второй класс продуктов для наклеек, в соответствии с табл. 10.3.

Таблица 10.3. Добавление класса наклеек

Параметр	Значение
Class ID (Идентификатор класса)	sticker
Class Name (Название класса)	Sticker (Наклейка)
Description (Описание)	A sticker with a witty phrase (Наклейка с остроумной фразой)

Настройка атрибутов продуктов

Интернет-магазин Sweet Tees должен предоставить покупателям возможность выбирать размер футболки, а также ее цвет и наклейку. Для этого мы задействуем атрибуты продуктов, реализованные в пакете Ubercart. На рис. 10.17 показан пример определения атрибута Size (Размер).

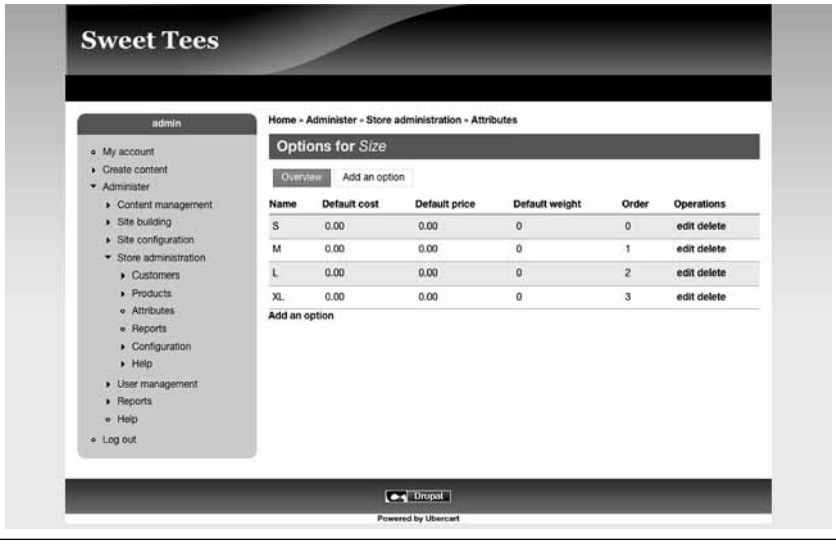


Рис. 10.17. *Атрибут Size (Размер) и его возможные значения*

1. Перейдите на страницу Administer (Администрирование)→Store administration (Управление магазином)→Attributes (Атрибуты) (*admin/store/attributes*) и щелкните на вкладке Add an attribute (Добавить атрибут) (*admin/store/attributes/add*).

2. На этой странице можно создавать атрибуты продуктов и определять для них допустимые значения. Сначала нам необходимо добавить информацию об атрибуте. Введите в форму значения из табл. 10.4.

Таблица 10.4. Параметры атрибута Size (Размер)

Параметр	Значение
Name (Название)	Size (Размер)
Help text (Справочный текст)	Pick a T-shirt size (Укажите размер футболки)
Make this attribute required (Сделать этот атрибут обязательным)	отмечено
Display type (Тип отображения)	Select box (Список выбора)

3. Щелчок на кнопке Submit (Отправить) вернет нас обратно на вкладку Attributes Overview (Обзор атрибутов) (*admin/store/attributes*). Щелкните на ссылке options (значения) в колонке Operations (Операции) для атрибута Size (Размер).
4. В результате откроется страница просмотра допустимых значений, где будет представлен список всех возможных значений атрибута. Щелкните на вкладке Add an option (Добавить значение) (*admin/store/attributes/1/options/add*).
5. Следующая форма позволит добавить значения атрибута Size (Размер). Заполните форму значениями из табл. 10.5. Обратите внимание, что вы также можете указать Cost (Стоимость), Price (Цена) и Weight (Вес) для каждого значения атрибута. В нашем случае будем полагать, что стоимость и вес футболок не зависит от размера.

Таблица 10.5. Значения атрибута Size (Размер)

Название	Номер по порядку
S	0
M	1
L	2
XL	3

6. Теперь создадим атрибут цвета для футболок и наклеек. Вернитесь на страницу Administer (Администрирование)→Store administration (Управление магазином)→Attributes (Атрибуты) (*admin/store/attributes*) и щелкните на вкладке Add an attribute (Добавить атрибут) (*admin/store/attributes/add*). Введите значения из табл. 10.6.

Таблица 10.6. Параметры атрибута Color (Цвет)

Параметр	Значение
Name (Название)	Color (Цвет)
Help text (Справочный текст)	Select a color (Выберите цвет)
Make this attribute required (Сделать этот атрибут обязательным)	отмечено
Display type (Тип отображения)	Select box (Список выбора)

7. Вернувшись на страницу обзора, щелкните на ссылке options (значения) для атрибута Color (Цвет), щелкните на вкладке Add an option (Добавить значение) и введите значения из табл. 10.7.

Таблица 10.7. Значения атрибута Color (Цвет)

Название	Номер по порядку
White (Белый)	–1
Dark blue (Темно-синий)	0
Light blue (Светло-синий)	0
Plaid (В клетку)	0

8. Теперь необходимо связать новые атрибуты с классами продуктов. Вернитесь на страницу Administer (Администрирование)→Store administration (Управление магазином)→Products (Продукты)→Manage classes (Управление классами) (*admin/store/products/classes*) и щелкните на ссылке edit (изменить) в колонке Operation (Операция) для продукта T-shirt (Футболка) (*admin/store/products/classes/tshirt/edit*).
9. Щелкните на вкладке Attributes (Атрибуты) (*admin/store/products/classes/tshirt/attributes*) и затем на ссылке add attributes to this class (добавить атрибуты для этого класса) в справочном тексте (*admin/store/products/classes/tshirt/attributes/add*).
10. В списке Attributes (Атрибуты) выберите атрибуты Size (Размер) и Color (Цвет) и щелкните на кнопке Add attributes (Добавить атрибуты).
11. Щелкните на вкладке Options (Значения) (*admin/store/products/classes/tshirt/options*).
12. Выберите все четыре цвета и все четыре размера, отметив флажки для каждого значения. Выберите цвет White (Белый), как цвет по умолчанию, и размер M, как размер по умолчанию. Теперь страница с настройками должна выглядеть, как изображено на рис. 10.18. По окончании щелкните на кнопке Submit (Отправить).

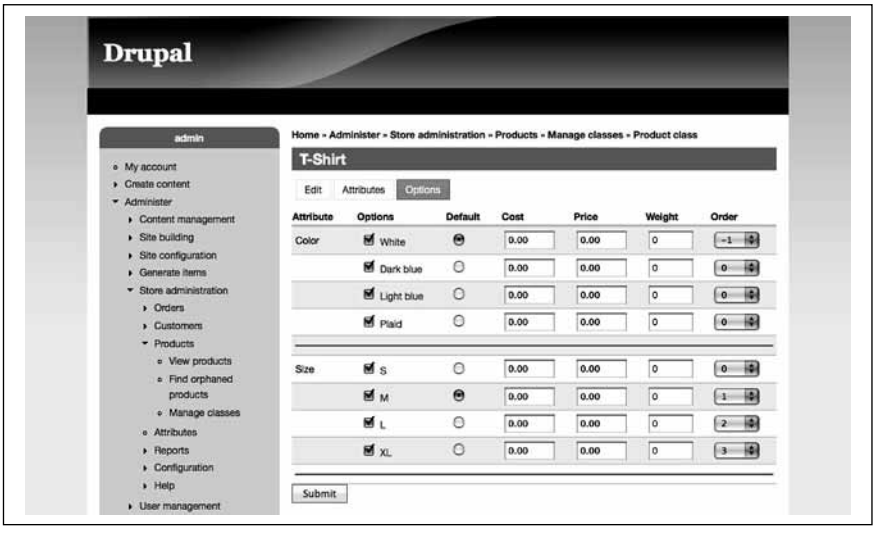


Рис. 10.18. *Значения атрибутов для футболок*

13. Наклейки, в отличие от футболок, не имеют размера и могут иметь только белый или светло-синий цвет. Вернитесь на страницу Administer (Администрирование)→Store administration (Управление магазином)→Products (Продукты)→Manage classes (Управление классами) (*admin/store/products/classes*) и затем щелкните на ссылке edit (изменить) для класса Sticker (Наклейка) (*admin/store/products/classes/sticker/edit*).
14. Щелкните на вкладке Attributes (Атрибуты) (*admin/store/products/classes/sticker/attributes*) и щелкните на ссылке add attributes to this class (добавить атрибуты для этого класса) в справочном тексте (*admin/store/products/classes/sticker/attributes/add*). В списке Attributes (Атрибуты) выберите пункт Color (Цвет) и щелкните на кнопке Add attributes (Добавить атрибуты).
15. Щелкните на вкладке Options (Значения) (*admin/store/products/classes/sticker/options*) и отметьте цвета White (Белый) и Light blue (Светло-синий). Сделайте цвет White (Белый) цветом по умолчанию и щелкните на кнопке Submit (Отправить).

Настройка параметров продуктов

Прежде чем перейти к созданию каталога товаров, настроим некоторые основные параметры продуктов:

1. Перейдите на страницу Administer (Администрирование)→Store administration (Управление магазином)→Configuration (Настройка)→Product settings (Параметры продуктов) (*admin/store/settings/products*) и щелкните на заголовке раздела Product settings (Параметры продуктов).

2. Эта страница позволяет изменить некоторые значения по умолчанию параметров продуктов, таких как количество отображаемых записей о продуктах в списках и порядок обработки ссылок Add to cart (Добавить в корзину). Значения по умолчанию нас вполне устраивают; единственное, что нужно изменить, – это отметить флажок Display an optional quantity field in the Add to Cart form (Отображать количество товаров в форме «Добавить в корзину»). Щелкните на кнопке Save configuration (Сохранить конфигурацию).
3. Щелкните на вкладке Product fields (Поля продукта) (*admin/store/settings/products/edit/fields*). Здесь мы можем выбирать дополнительные поля для отображения в описаниях всех продуктов. В нашем случае эти поля являются общими для всех продуктов. В дополнение к полям, выбранным по умолчанию, установите флажок для поля Weight (Вес) и щелкните на кнопке Save configuration (Сохранить конфигурацию).

Так как продукты Ubercart являются типами содержимого Drupal, имеется возможность добавлять в них дополнительные поля CCK, как для любого другого типа содержимого, перейдя на страницу Administer (Администрирование)→Content management (Содержание)→Content types (Типы материалов) (*admin/content/types*) и щелкнув на ссылке manage fields (управлять полями) для типа Product (Продукт). Поля, определяемые пакетом Ubercart, являются общими для всех продуктов.

Настройка каталога

Модуль Catalog, включенный ранее, позволяет просматривать продукты по категориям, как показано на рис. 10.19. Весь механизм каталога построен на базе механизма таксономии Drupal, поэтому добавление и управление иерархией каталога осуществляется с помощью стандартного интерфейса управления таксономией. Однако пакет Ubercart добавляет несколько интересных возможностей, позволяющих просматривать и получать списки продуктов по категориям.

Прежде чем приступать к чтению следующего раздела, убедитесь, что словарь Catalog (Каталог) был создан, проверив сообщения о состоянии на странице Administer (Администрирование)→Store administration (Управление магазином) (*admin/store*). Если все в порядке, вы должны увидеть сообщение «Vocabulary Catalog has been identified as the Ubercart catalog» (Словарь Catalog был идентифицирован, как каталог Ubercart). Это означает, что словарь каталога был благополучно создан. Если это сообщение отсутствует, создайте новый словарь вручную, перейдите на страницу Administer (Администрирование)→Store administration (Управление магазином)→Configuration (Настройка)→Catalog settings (Параметры каталога) (*admin/store/settings/catalog*) и отметьте его, как словарь каталога.

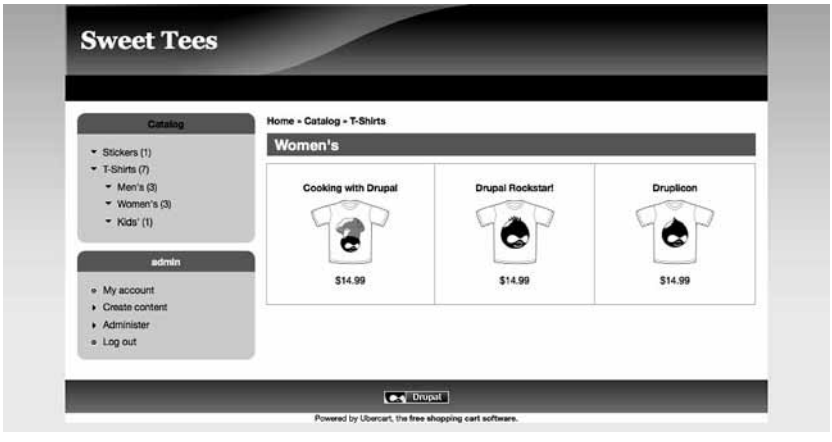


Рис. 10.19. *Каталог товаров на сайте Sweet Tees*

1. Для начала добавим несколько категорий в каталог. Перейдите на страницу Administer (Администрирование)→Content management (Содержание)→Taxonomy (Таксономия) (*admin/content/taxonomy*).
2. Щелкните на ссылке add terms (добавить термины) для словаря Catalog (Каталог) и добавьте категории продуктов, перечисленные ниже. С помощью дополнительных параметров Parents (Родители) и Weight (Вес) расположите термины в следующем порядке:
   - Stickers (Наклейки)
   - T-shirts (Футболки)
     – Men's (Мужские)
     – Women's (Женские)
     – Kids' (Детские)
3. Перейдите на страницу Administer (Администрирование)→Store administration (Управление магазином)→Configuration (Настройка)→Catalog settings (Параметры каталога) (*admin/store/settings/catalog*).
4. Затем щелкните на заголовке раздела Catalog (Каталог) (*admin/store/settings/catalog/edit/catalog*), чтобы открыть форму редактирования. Обратите внимание, что у вас имеется возможность выбирать любой словарь таксономии, который будет использоваться как словарь каталога (если у вас уже есть словарь, который вы хотели бы задействовать).
5. Большую часть параметров мы оставим со значениями по умолчанию. Измените настройки в соответствии с табл. 10.8 и щелкните на кнопке Save configuration (Сохранить конфигурацию).

Таблица 10.8. Параметры отображения каталога

Параметр	Значение
Catalog products list	
Product nodes per page (Продуктов на странице)	12
Catalog block settings	
Always expand categories in the catalog block (Всегда разворачивать категории каталога в блоке)	отмечено

6. Щелкните на вкладке Grid (Сетка) (*admin/store/settings/catalog/edit/grid*).
7. Здесь мы можем расположить страницы нашего каталога не в виде обычного списка, а в виде сетки. Заполните форму в соответствии со значениями в табл. 10.9 и щелкните на кнопке Save configuration (Сохранить конфигурацию).

Таблица 10.9. Настройка отображения каталога в виде сетки

Параметр	Значение
Display products in grid (Отображать продукты в виде сетки)	отмечено
Display product model (SKU) (Отображать модель продукта (артикул))	не отмечено
Display product add to cart (Отображать ссылку «добавить в корзину»)	не отмечено

8. Теперь добавим в каталог несколько продуктов. Перейдите на страницу Create Content (Создать материал)→T-shirt (Футболка) (*node/add/tshirt*), заполните форму в соответствии с табл. 10.10 и щелкните на кнопке Save (Сохранить), чтобы создать продукт. Изображения футболок можно найти в исходных кодах, распространяемых вместе с книгой, в каталоге *assets/ch10-store*; кроме того, вы можете использовать свои изображения. По окончании описание продукта будет выглядеть, как показано на рис. 10.20.

Таблица 10.10. Форма создания продукта T-shirt (Футболка)

Поле	Значение
Name (Название)	Druplicon (Футболка Drupal)
Catalog	Men's (Мужские), Women's (Женские)

Таблица 10.10. Форма создания продукта T-shirt (Футболка) (Продолжение)

Поле	Значение
SKU	T-shirt001
Sell price (Цена)	14.99
Weight (Вес)	2 фунта (900 грамм)
Описание	Drupal's logo (Футболка с логотипом Drupal)
Image (Изображение)	[выгрузите изображение футболки]

Если поле изображения не отображается здесь, проверьте еще раз сообщения о состоянии на главной странице административного раздела Ubercart Administer (Администрирование)→Store administration (Управление магазином) (*admin/store*) и выполните инструкции для изображений. Возможно, ранее вы забыли включить некоторые из необходимых модулей.

Рис. 10.20. *Пример продукта «Футболка»*

9. Добавьте еще несколько продуктов, чтобы немного наполнить каталог. Обратите внимание, что каждый продукт может иметь собственные атрибуты, их значения и корректировки. Вы можете изменять их, щелкнув на вкладке Edit (Изменить) и затем на соответствующих вкладках, расположенных под вкладками View (Просмотр) и Edit (Изменить). Например, если на складе отсутствуют футболки определенных размеров, можно щелкнуть на вкладке Options (Значения) и снять флажки для отсутствующих размеров.

10. Теперь посмотрим, что у нас получилось. Включите каталог, перейдя на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Blocks (Блоки) (*admin/build/block*) и отбуксируйте блок Catalog (Каталог) в начало области left sidebar (левая колонка). Затем щелкните на кнопке Save blocks (Сохранить блоки). Теперь мы можем просматривать каталог товаров, щелкая на ссылках в блоке Catalog (Каталог), находящемся в боковой колонке.
11. В заключение настроим разрешения на доступ к созданным нами элементам. Перейдите на страницу Administer (Администрирование)→User management (Управление пользователями)→Permissions (Разрешения) (*admin/user/permissions*), выполните настройки в соответствии с табл. 10.11 и щелкните на кнопке Save permissions (Сохранить права доступа).

Таблица 10.11. Разрешения для компонентов интернет-магазина

Разрешение	**anonymous user (анонимный пользователь)**	**authenticated user (зарегистрированный пользователь)**	**editor (редактор)**	**site administrator (администратор сайта)**
модуль uc_catalog				
administer catalog (управлять каталогом)				отмечено
view catalog (просматривать каталог)	отмечено	отмечено		
модуль uc_product				
administer product classes (управлять классами продуктов)				отмечено
administer products (управлять продуктами)				отмечено
create products (создавать продукты)			отмечено	
edit all products (изменять все продукты)			отмечено	

Разрешение	anonymous user (анонимный пользователь)	authenticated user (зарегистрированный пользователь)	editor (редактор)	site administrator (администратор сайта)
модуль uc_store				
administer store (управлять магазином)				отмечено
view customers (просматривать информацию о клиентах)			отмечено	отмечено
view store reports (просматривать отчеты)			отмечено	отмечено

В центре внимания: процесс оформления заказа

Теперь у нас имеется интернет-магазин, который можно наполнить сведениями о товарах, имеющихся у фирмы Sweet Tees. Однако в настоящий момент покупатели могут только просматривать каталог и получать информацию о продуктах. Большинство сайтов электронной коммерции занимаются фактической продажей каких-либо товаров, а это означает наличие процедуры оформления заказа.

На рис. 10.21 изображен типичный процесс покупки для интернет-магазина, такого как наш. Процесс покупки начинается с того, что покупатель добавляет продукты в электронную покупательскую корзину и щелкает на кнопке Checkout (Купить). Перед покупателем открывается форма, в которой он должен указать основную информацию о себе, такую как расчетный адрес и адрес доставки, информация о кредитной карте и предпочтительный способ доставки. После этого покупатель получает возможность просмотреть заказ, включая общую стоимость, сумму налогов и стоимость доставки, вычисленные на основе информации, предоставленной ранее. Как только заказ будет отправлен, система электронных платежей проверит введенную учетную информацию, чтобы убедиться в правильности номера кредитной карты. Если никаких проблем не возникло, заказ принимается и магазин исполняет свою часть сделки, упаковывая товары и отправляя по указанному адресу.

Несмотря на то что это типичный процесс для «обычных» интернет-магазинов, разные магазины имеют совершенно разные потребности. Пакет модулей Ubercart позволяет определять самые разные способы оформления заказов. Желаете продавать загружаемые продукты, кото-

рые не требуется доставлять куда-либо? Нет проблем. Включите модуль Downloads и отключите модуль Shipping. Вам необходимо использовать сложные международные правила исчисления налогов? Эту потребность удовлетворит модуль Conditional Actions в комбинации с модулем Taxes. Процесс оформления заказа в вашей системе требует выполнения большого числа операций, таких как «Ожидание получения различных документов» или «Необходимость получения счета-фактуры»? Модуль Orders дает возможность определять требуемые состояния процесса оформления заказа, благодаря которым вы всегда будете знать, на какой стадии находится открытый заказ.

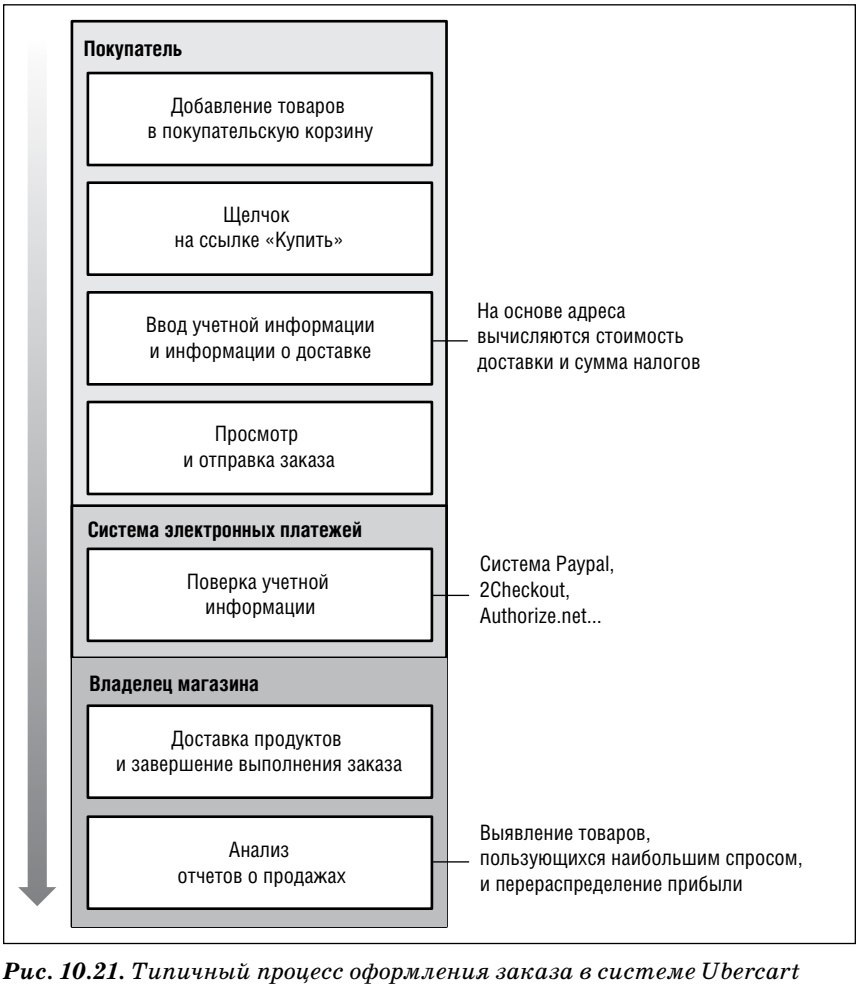


Рис. 10.21. *Типичный процесс оформления заказа в системе Ubercart*

Параметров настройки слишком много, чтобы подробно обсуждать их здесь, поэтому мы рассмотрим лишь некоторые группы настроек, непосредственно касающиеся процедуры оформления заказа, которые мож-

но найти на странице Administer (Администрирование)→Store administration (Управление магазином)→Configuration (Настройка) (*admin/store/settings*):

Cart settings (Параметры корзины)

Эти параметры определяют поведение системы в процессе покупки, например адрес, куда должен быть перенаправлен покупатель после добавления продуктов в свою корзину и какую информацию следует отображать на странице покупательской корзины

Order settings (Параметры оформления заказов)

Эти параметры определяют внешний вид формы заказа, различные состояния заказа и порядок отображения информации о заказе в административной панели

Checkout settings (Параметры покупки)

Эти параметры определяют перечень видимых полей на странице оформления покупки и текст различных системных сообщений

Payment settings (Параметры оплаты)

Эти параметры определяют порядок слежения за платежами, допустимые формы оплаты и настройки доступа к используемой системе электронных платежей

Shipping quote settings (Параметры вычисления стоимости доставки)

Эти параметры определяют заданные по умолчанию адрес и методы доставки

Tax rates and settings (Ставки налогов и настройки)

На этой странице можно создавать специфичные правила исчисления налогов

Скорее всего, вас вполне устроят значения по умолчанию, но определенно стоит потратить некоторое время, чтобы ознакомиться с некоторыми из этих параметров, так как вполне возможно, что они понадобятся вам в будущем.

Практика: обработка заказов

Последнее, что осталось сделать на нашем сайте, – внедрить элементы электронной коммерции: покупательскую корзину и обработку заказов, а также инструменты получения отчетов, которые позволят получать представление о деятельности магазина. На этом мы завершим настройку нашего магазина.

Для начала нам необходимо включить последний набор модулей. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Modules (Модули) (*admin/build/modules*) и включите следующие модули:

- Ubercart – core (Ubercart – ядро)
  - Cart
  - Conditional Actions
  - Order
- Ubercart – core (optional) (Ubercart – ядро (дополнительные))
  - Payment
  - Reports
  - Shipping
  - Shipping Quotes
  - Taxes
- Ubercart – fulfillment (Ubercart – исполнение)
  - Flatrate
- Ubercart – payment (Ubercart – платежи)
  - Credit Card
  - Payment Method Pack
  - Test Gateway

Точные инструкции по реализации системы оплаты в значительной степени зависят от используемых служб (Paypal, Authorize.net и так далее), поэтому в данной главе мы настроим проверку кредитных карт только с использованием фиктивного интерфейса обработки платежей Test Gateway. На «действующем» сайте электронной коммерции вам придется использовать фактическую систему электронных платежей, которая может принимать и обрабатывать кредитные карты. На веб-сайте проекта Ubercart имеется список поддерживаемых систем электронных платежей и руководство по их настройке: *http://www.ubercart.org/payment*.

Покупательская корзина

В этом разделе мы настроим покупательскую корзину, изображенную на рис. 10.22. После настройки корзины мы сможем просматривать отдельные страницы с описаниями футболок, определять такие характеристики, как цвет и количество, и добавлять выбранные продукты в корзину.

Большая часть работы по реализации корзины уже была выполнена простым включением модуля Cart. Корзина имеет вполне работоспособные настройки по умолчанию, но имеется пара параметров, которые нам необходимо изменить:

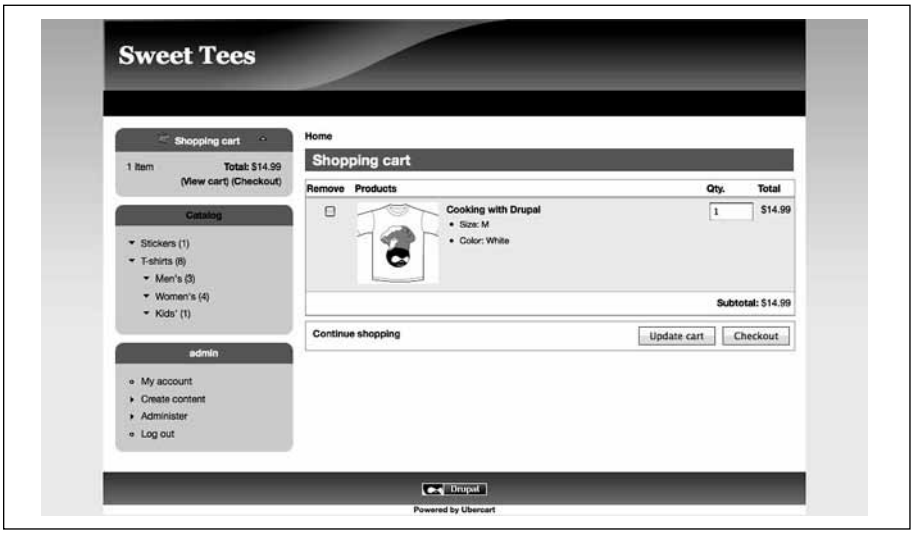


Рис. 10.22. *Покупательская корзина Ubercart и блок в боковой колонке*

1. Перейдите на страницу Administer (Администрирование)→Store administration (Управление магазином)→Configuration (Настройка)→Cart settings (Параметры корзины) (*admin/store/settings/cart*) и щелкните на заголовке раздела Cart settings (Параметры корзины) (*admin/store/settings/cart/edit*).
2. По умолчанию настройка параметра Continue shopping link URL (Адрес URL для ссылки «Продолжить покупки») возвращает пользователя на главную страницу сайта. Мы должны изменить этот параметр, чтобы направлять пользователя в каталог товаров. Введите значение «catalog» и щелкните на кнопке Save configuration (Сохранить конфигурацию).
3. Мы также создадим блок Shopping Cart (Покупательская корзина), чтобы у покупателя всегда имелся доступ к своей корзине на любой странице сайта. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Blocks (Блоки) (*admin/build/block*).
4. Отбуксируйте блок Shopping Cart (Покупательская корзина) в начало области left sidebar (левая колонка) и щелкните на кнопке Save blocks (Сохранить блоки).

Теперь наша покупательская корзина готова к работе. Это было совсем не сложно! Далее мы поговорим о том, что происходит, когда покупатель щелкает на кнопке Checkout (Купить).

Налоги

Прежде чем можно будет распахнуть двери нашего магазина перед покупателями, мы должны обеспечить применение всех налогов с продаж к нашим товарам. Так как фирма Sweet Tees находится в штате Калифорния, мы должны прибавлять к стоимости товаров налог на продажу в кредит.

Этот пример создан исключительно в демонстрационных целях: вы должны сами определить, какими налогами должны облагаться товары, продаваемые в вашем магазине. Модуль Conditional Actions, входящий в состав ядра пакета Ubercart, позволяет устанавливать разные правила налогообложения любой сложности для разных продуктов и при исчислении налогов учитывать, находится ли покупатель в том же штате, что и вы, в каком-то другом штате или за рубежом.

1. Перейдите на страницу Administer (Администрирование)→Store administration (Управление магазином)→Configuration (Настройка)→Tax rates and settings (Налоговые ставки и настройки) (*admin/store/settings/taxes*) и щелкните на ссылке Make a new tax rule (Создать новое правило налогообложения) (*admin/store/settings/taxes/edit*).
2. Заполните форму Edit tax rule (Редактирование правил налогообложения) в соответствии с табл. 10.12.

Таблица 10.12. Форма редактирования правил налогообложения

Поле	Значение
Name (Название)	Sales tax (Налог на добавленную стоимость)
Rate (Ставка)	7.25%
Type product types (Типы товаров, облагаемые налогом)	T-shirt (Футболка), Sticker (Наклейка)

3. Щелкните на кнопке Submit (Отправить), чтобы сохранить настройки налога на добавленную стоимость.

Доставка

Мы продаем физически существующие товары, или товары, которые требуют доставки, поэтому в цене товара нам необходимо учесть стоимость его доставки. Для простоты мы будем использовать единый тариф, предположив отсутствие базовой составляющей в стоимости доставки и установив стоимость доставки любых футболок равной $5.99.

1. Перейдите на страницу Administer (Администрирование)→Store administration (Управление магазином)→Configuration (Настройка)→Shipping quote settings (Параметры вычисления стоимости доставки) (*admin/store/settings/quotes*) и щелкните на заголовке раздела Quote methods (Метод исчисления) (*store/settings/quotes/methods*).
2. Щелкните на вкладке Flat rate (Единый тариф) (*admin/store/settings/quotes/methods/flatrate*) и затем на ссылке Add a new flat rate shipping method (Добавить новый способ доставки по единому тарифу).
3. Заполните форму в соответствии с табл. 10.13.

Таблица 10.13. Настройки вычисления стоимости доставки по единому тарифу

Поле	Значение
Shipping method title (Название способа доставки)	Default shipping (Способ по умолчанию)
Line item label (Название позиции)	Shipping (Доставка)
Base price (Базовая цена)	0.00
Default product shipping rate (Тариф по умолчанию доставки товара)	5.99

Оплата

Теперь можно настроить заключительный элемент нашего сайта: систему оплаты. Пакет Ubercart обеспечивает большое количество вариантов приема платежей. Мы сможем принимать чеки и денежные переводы, а также принимать оплату кредитными картами (с помощью фиктивного интерфейса).

1. Перейдите на страницу Administer (Администрирование)→Store administration (Управление магазином)→Configuration (Настройка)→Payment settings (Параметры оплаты) (*admin/store/settings/payment*) и щелкните на заголовке раздела Payment methods (Способы оплаты) (*admin/store/settings/payment/edit/methods*).
2. В таблице Payment methods (Способы оплаты) отметьте варианты Check (Чек) и Credit card (Кредитная карта) и в качестве системы приема платежей выберите Test Gateway (Фиктивный интерфейс). Щелкните на кнопке Save configuration (Сохранить конфигурацию).
3. В верхней части страницы должно появиться сообщение об ошибке: «Credit card encryption must be configured to accept credit card payments» (Для приема платежей по кредитным картам необходимо определить метод шифрования). Исправим эту проблему.

При приеме платежей по кредитным картам через Интернет необходимо принять во внимание несколько проблем, связанных с обеспечением безопасности. Для приема информации вы всегда должны использовать заверенный сертификат SSL и, если это возможно, не сохранять номера кредитных карт на сайте. За дополнительной информацией по обеспечению безопасности операций с кредитными картами в Ubercart обращайтесь к документации: *http://www.ubercart.org/docs/user/2731/credit_card_settings#security*.

4. Создайте в файловой системе каталог с именем *keys*, который будет использоваться для хранения данных о кредитных картах в зашифрованном виде. Этот каталог *должен быть недоступен для веб-сервера*. Например, если главная страница веб-сайта находится в каталоге */home/username/www*, создайте каталог */home/username/keys*.
5. Временно сделайте каталог доступным для записи, например, командой `chmod a+w /home/username/keys`.
6. Разверните группу параметров Credit card settings (Параметры кредитных карт) и в параметре Card number encryption key filepath (Путь к ключу шифрования кредитных карт) введите путь к файлу ключа, например */home/username/keys*.
7. В разделе Checkout workflow (Процесс покупки) установите флажок Attempt to process credit card payments at checkout (Пытаться обрабатывать платежи по кредитным картам).
8. Разверните группу параметров Check settings (Параметры оплаты чеками) и введите какой-нибудь адрес магазина.
9. Щелкните на кнопке Save configuration (Сохранить конфигурацию).
10. В файловой системе опять сделайте каталог недоступным для записи, например, командой `chmod a-w /home/username/keys`.

Размещение тестового заказа

Теперь мы готовы сделать наш первый заказ! Ниже описывается, как это можно сделать:

1. Отыщите в каталоге наш продукт «T-shirt001». Мы должны выбрать цвет и размер и затем щелкнуть на ссылке Add to cart (Добавить в корзину). После этого откроется представление корзины. Обратите внимание, что блок Shopping cart (Покупательская корзина) немедленно отобразит количество товаров (1) и общую стоимость выбранных товаров.
2. Находясь в представлении покупательской корзины, можно щелкнуть на ссылке Continue shopping (Продолжить покупки) и вернуться обратно в каталог или выполнить операции с товарами в корзи-

не, удалив ненужные или изменив количество любого товара (не забыв после изменений щелкнуть на кнопке Update cart (Обновить корзину)). Однако наша задача – проверить процедуру покупки в действии, поэтому щелкните на кнопке Checkout (Купить).

3. Теперь мы попали на страницу оформления покупки. В верхней части страницы, как показано на рис. 10.23, для проверки отображается содержимое нашей корзины и ниже следует раздел ввода информации о покупателе, где отображается наш адрес электронной почты, так как мы выполняем покупку, предварительно зарегистрировавшись на сайте.

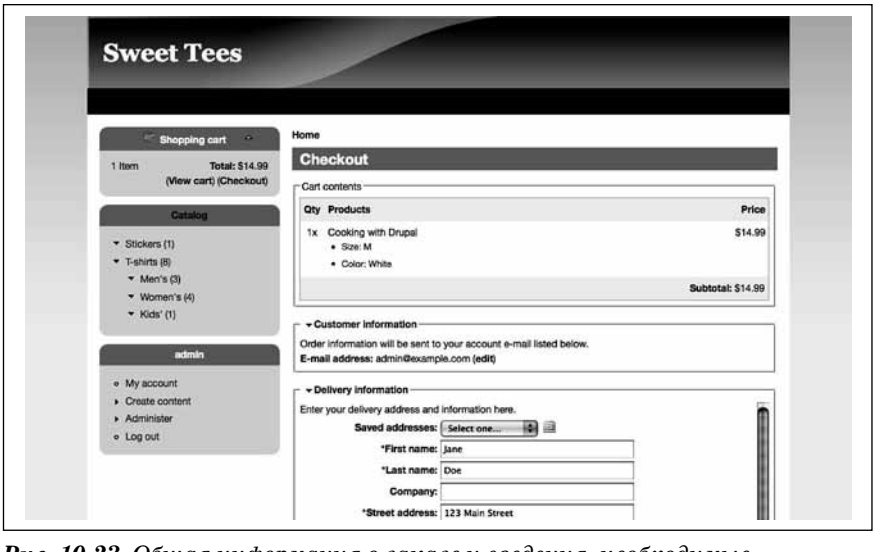


Рис. 10.23. *Общая информация о заказе и сведения, необходимые для доставки товара, на странице оформления покупки*

4. Заполните разделы с информацией о доставке и с учетной информацией.
5. Ниже находится раздел, позволяющий выбрать способ оплаты. По умолчанию выбрана радиокнопка Check or money order (Чек или наличные) и указан адрес, введенный нами ранее. Щелкните на радиокнопке Credit card (Кредитная карта), после чего раздел автоматически обновится и появится форма ввода информации о кредитной карте, как показано на рис. 10.24.
6. Мы используем фиктивный интерфейс связи с системой электронных платежей, поэтому от нас не требуется вводить действительный номер кредитной карты. Однако информация должна вводиться в допустимом формате. Модуль Credit Card проверяет количество цифр в номере кредитной карты и срок ее действия. Заполните поля этого раздела в соответствии с табл. 10.14.

Таблица 10.14. Форма с информацией о кредитной карте

Поле	Значение
Card Number	4111111111111111 (здесь 15 единиц)
Expiration Date	July 2017 (июль 2017)
CVV (код проверки подлинности карты)	123

7. После щелчка на кнопке Review order (Просмотреть заказ) откроется страница подтверждения, изображенная на рис. 10.25, для просмотра всей введенной информации. Если здесь будут обнаружены ошибки, щелкните на кнопке Back (Назад), и вы вернетесь на страницу оформления покупки, где можно будет внести исправления.

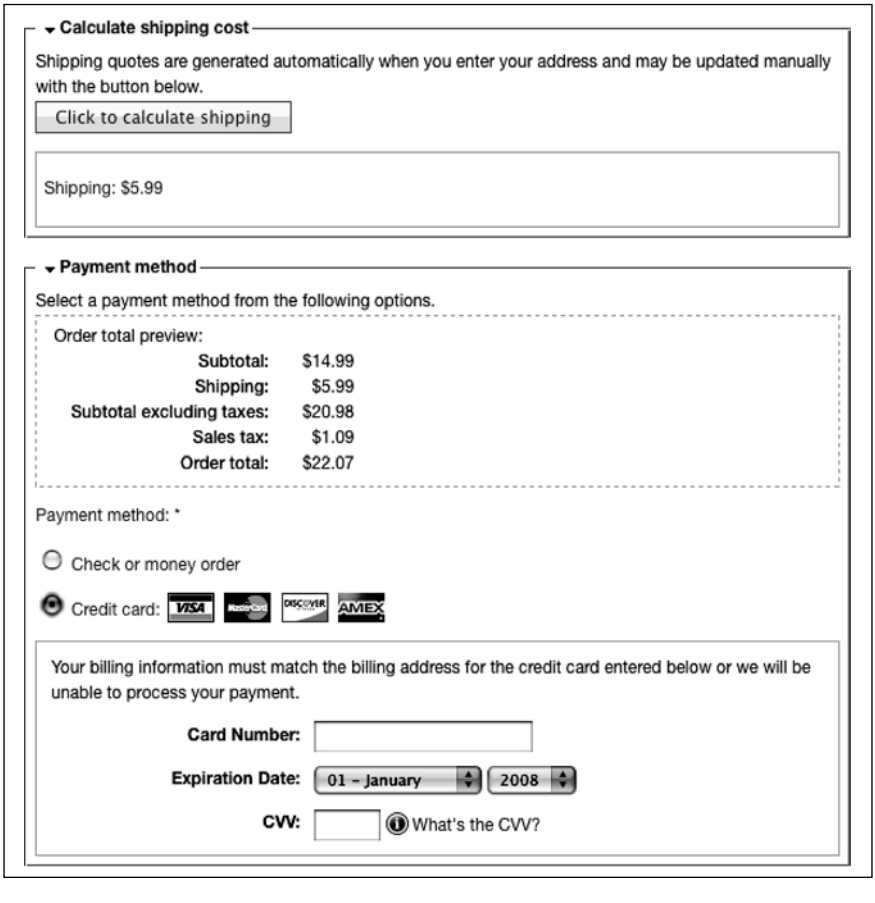


Рис. 10.24. *Раздел с информацией о способе доставки и оплаты*

Щелчок на кнопке Submit order (Отправить заказ) завершит операцию покупки, при этом по электронной почте вам будет отправлено подтверждение и будет создана запись о заказе для администраторов интернет-магазина.

8. По окончании перед покупателем будет открыта страница с благодарностью за покупку и со ссылкой для просмотра текущего состояния заказа в разделе Orders (Заказы) в профиле данного пользователя.

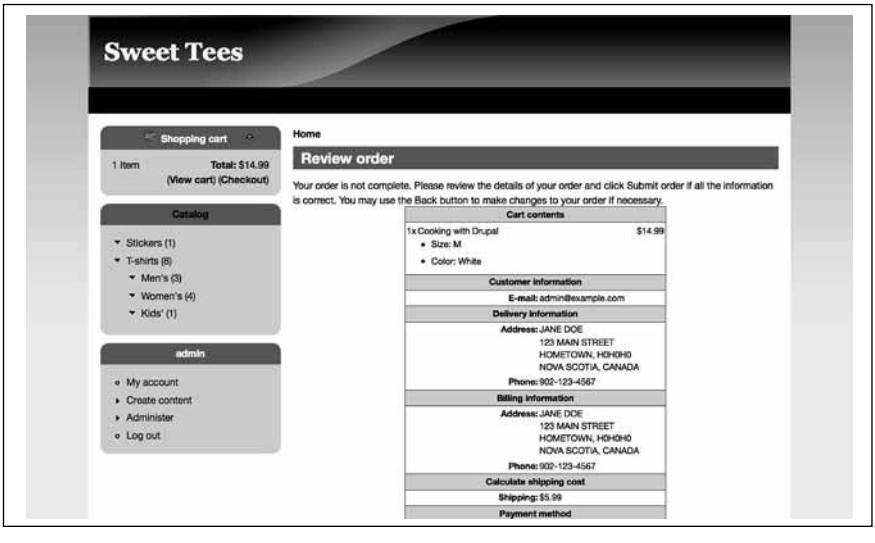


Рис. 10.25. *Страница просмотра заказа, где покупатель может сделать заключительные корректировки*

Исполнение заказа

Мы благополучно разместили наш тестовый заказ. Теперь посмотрим, что происходит дальше: как производится исполнение заказа, отгрузка товаров и просмотр отчетов о деятельности магазина.

1. Перейдите на страницу Administer (Администрирование)→Store administration (Управление магазином)→Orders (Заказы) (*admin/store/orders*), чтобы увидеть список всех заказов в системе. Наш заказ отображается с пометкой «Payment received» (Оплачено), потому что фиктивный интерфейс системы электронных платежей отработал должным образом. Щелкните на ярлыке View (Просмотреть), чтобы перейти на страницу с информацией о заказе, которая должна выглядеть, как показано на рис. 10.26. На этой странице можно выполнить такие операции, как распечатка счета или отправка его по почте, можно просмотреть журнал изменений заказа и подробности об оплате.

2. Отсюда также можно инициировать операцию отгрузки товара, о которой мы поговорим ниже. Щелкните на вкладке Packages (Посылки)

и затем на ссылке Create packages (Создать посылки), чтобы открыть страницу, изображенную на рис. 10.27.

Рис. 10.26. *Просмотр информации о заказе*

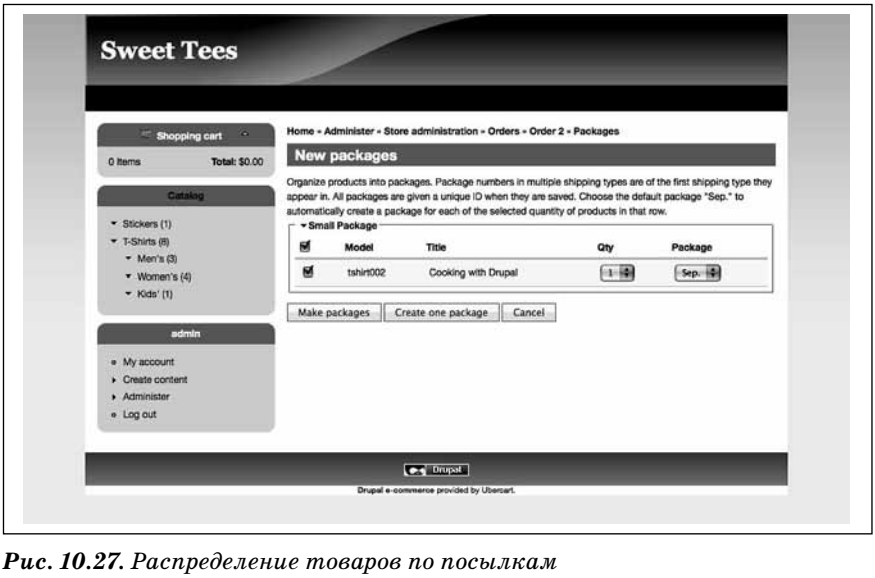


Рис. 10.27. *Распределение товаров по посылкам*

3. Отметьте строку с заказом футболки и щелкните на кнопке Create one package (Создать одну посылку). На этой странице можно также, например, организовать отправку двух товаров одной посылкой, а третьего – другой.

4. Затем щелкните на вкладке Shipments (Отгрузки) и на ссылке Make a new shipment (Создать новую отгрузку). Отметьте посылку, выберите значение Ship manually (Отгрузить вручную) в поле способа отгрузки и щелкните на кнопке Ship packages (Отгрузить посылки).
5. После этого откроется форма, где можно ввести дополнительную информацию, такую как номер слежения, дата отгрузки и дата доставки. Просто щелкните на кнопке Save shipment (Сохранить информацию об отгрузке), чтобы принять значения по умолчанию.
6. В заключение мы должны изменить заказ, чтобы отметить, что заказ выполнен и благополучно отправлен. Щелкните на вкладке View (Просмотреть), чтобы вернуться к странице представления заказа.
7. Под заказом имеется раскрывающийся список, определяющий состояние заказа. Выберите в списке значение Completed (Выполнено) и щелкните на кнопке Update (Обновить).
8. Теперь, когда мы завершили выполнение нашего первого заказа, самое время заглянуть в систему отчетов, предоставляемую пакетом Ubercart. Перейдите на страницу Administer (Администрирование)→Store administration (Управление магазином)→Reports (Отчеты) (*admin/store/reports*). Щелкните на ссылке Sales reports (Отчеты о продажах), чтобы посмотреть сводный отчет о деятельности магазина, изображенный на рис. 10.28. Не стесняйтесь заглянуть и в другие отчеты, имеющиеся в этом разделе.

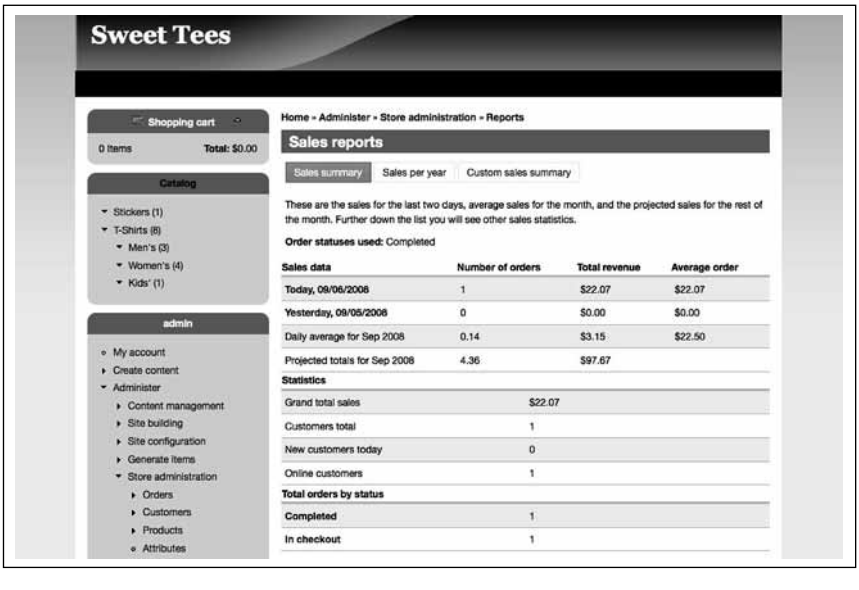


Рис. 10.28. *Отчеты о продажах показывают общее состояние дел в магазине*

Управление доступом

В заключение настроим права доступа к новым модулям, которые мы включили в этом разделе. Перейдите на страницу Administrator (Администрирование)→User management (Управление пользователями)→ Permissions (Разрешения) (*admin/user/permissions*). Установите разрешения в соответствии с табл. 10.15 и щелкните на кнопке Save permissions (Сохранить права доступа). Большая часть этих разрешений уже знакома нам, или об их назначении можно догадаться по названиям, но некоторые из них заслуживают особого внимания, так как из их названий трудно догадаться, какие последствия они могут иметь:

- В разделе *модуль uc_credit* присутствуют два разрешения, касающиеся кредитных карт: view cc details (просмотр информации о кредитных картах) и view cc numbers (просмотр номеров кредитных карт). Разрешение view cc details (просмотр информации о кредитных картах) дает пользователям возможность узнать тип использовавшейся кредитной карты и сумму платежа, поэтому мы дадим это разрешение роли *editor*. Однако номер кредитной карты относится к разряду секретной информации. Даже при том, что Ubercart обычно сохраняет только последние четыре цифры номера кредитной карты, лучше перестраховаться, чем потом сожалеть. Мы дадим это разрешение только администраторам сайта.
- В разделе *модуль uc_order* присутствуют два разрешения с похожими названиями: delete orders (удалять заказы) и delete any orders (удалять любые заказы). Разница заключается в том, что пользователи, обладающие разрешением delete any orders (удалять любые заказы), смогут удалять из системы даже завершенные заказы, минуя любые дополнительные проверки, которые могли бы воспрепятствовать удалению заказа. Поэтому мы дадим это разрешение только роли *site administrator*.

Таблица 10.15. Разрешения для модулей обработки заказов из пакета Ubercart

Разрешение	**anonymous user (анонимный пользователь)**	**authenticated user (зарегистрированный пользователь)**	**editor (редактор)**	**site administrator (администратор сайта)**
модуль ca				
administer conditional actions (управлять условными действиями)				отмечено

Таблица 10.15 (продолжение)

Разрешение	anonymous user (анонимный пользователь)	authenticated user (зарегистрированный пользователь)	editor (редактор)	site administrator (администратор сайта
модуль uc_credit				
administer credit cards (управлять кредитными картами)				отмечено
process credit cards (обрабатывать кредитные карты)			отмечено	отмечено
view cc details (просматривать информацию о кредитных картах)			отмечено	отмечено
view cc numbers (просматривать номера кредитных карт)				отмечено
модуль uc_order				
administer order workflow (управлять процессом оформления заказа)				отмечено
create orders (создавать заказы)			отмечено	отмечено
delete any order (удалять любые заказы)				отмечено
delete orders (удалять заказы)			отмечено	отмечено
edit orders (изменять заказы)			отмечено	отмечено

view all orders (просматривать все заказы)			отмечено	отмечено
модуль uc_payment				
delete payments (удалять информацию об оплате)				отмечено
manual payments (принимать платежи вручную)			отмечено	отмечено
view payments (просматривать платежи)			отмечено	отмечено
модуль uc_quote				
configure quotes (настраивать расценки)				отмечено
модуль uc_reports				
view reports (просматривать отчеты)			отмечено	отмечено
модуль uc_shipping				
fulfill orders (исполнять заказы)			отмечено	отмечено
модуль uc_store				
view store reports (просматривать отчеты магазина)			отмечено	отмечено
модуль uc_taxes				
configure taxes (настраивать взимание налогов)				отмечено

Дополнительно

В этой главе мы рассмотрели основы создания витрины и покупательской корзины для интернет-магазина с использованием пакета Ubercart

для системы Drupal. Однако имеется несколько дополнительных модулей, возможность применения которых вы наверняка захотели бы рассмотреть, прежде чем открыть свой интернет-магазин:

Secure Pages (http://drupal.org/project/securepages)

Если вы предусматриваете получение секретной частной информации от своих клиентов через Интернет, такой как номера кредитных карт, настоятельно рекомендуется, чтобы эти операции производились через защищенные соединения SSL. Модуль Secure Pages позволяет определить пути в системе Drupal, которые должны посещаться исключительно по протоколу HTTPS. Рекомендуется защищать такие пути, как *user*/* и *cart*/*.

PayPal (входит в состав пакета Ubercart)

Модуль PayPal не является обязательным к использованию, однако вам наверняка потребуется задействовать какую-либо систему электронных платежей (не забывайте, что в этой главе мы использовали модуль Test Gateway, реализующий интерфейс к фиктивной службе платежей). Службы системы PayPal просты в настройке и прекрасно поддерживаются пакетом Ubercart.

Stock (входит в состав пакета Ubercart)

При торговле товарами, такими как футболки, будет совсем не лишним организовать слежение за текущим уровнем запасов на складах, чтобы не продать кому-нибудь товар, отсутствующий на складе. Модуль Stock (находится в категории Ubercart – extra (Ubercart – дополнительные)) изменяет количество товаров на складе всякий раз, когда производится покупка. При получении новой партии товара просто добавьте его количество к текущему уровню запасов. Кроме того, модуль позволяет контролировать достижение порогового значения в количестве товара и инициировать отправку по электронной почте извещения об этом.

В заключение

В этой главе мы сумели построить законченный интернет-магазин для нашего клиента – фирмы Sweet Tees. Несмотря на огромное число модулей, страниц с настройками и способов переопределения значений, достаточно разумные значения по умолчанию и автоматизация решения типичных задач, таких как настройка наборов правил для модуля ImageCache, делают пакет Ubercart чрезвычайно простым в использовании.

Ниже приводится перечень модулей, упоминавшихся в этой главе:

- CCK: *http://drupal.org/project/cck*
- e-Commerce: *http://drupal.org/project/ecommerce*
- FileField: *http://drupal.org/project/filefield*

- Google Analytics: *http://drupal.org/project/google_analytics*
- ImageAPI: *http://drupal.org/project/imageapi*
- ImageCache: *http://drupal.org/project/imagecache*
- ImageField: *http://drupal.org/project/imagefield*
- Token: *http://drupal.org/project/token*
- Ubercart: *http://drupal.org/project/ubercart*

Ниже приводятся ссылки на дополнительные ресурсы, упоминавшиеся в этой главе:

- Официальный сайт проекта Ubercart: *http://www.ubercart.org*
- Сторонние модули для Ubercart: *http://www.ubercart.org/contrib*
- Статья об обеспечении безопасности при работе с кредитными картами: *http://www.ubercart.org/docs/user/2731/credit_card_settings#security*
- Системы электронных платежей, поддерживаемые пакетом Ubercart: *http://www.ubercart.org/payment*

11

Оформление сайта

В предыдущей части книги мы активно обсуждали вопросы конструирования самых различных веб-сайтов, от сайтов обмена фотографиями и каталогов продуктов до сайтов управления событиями, за счет комбинирования базовых возможностей системы Drupal с десятками дополнительных сторонних модулей. Система Drupal предоставляет в наше распоряжение множество инструментов реализации и настройки функциональности наших сайтов, но зачастую многие веб-сайты отличаются в основном внешним видом.

Если задуматься, то окажется, что сайты YouTube и Flickr не так сильно отличаются по своим функциональным возможностям. Безусловно, отличия есть, так как один из них занимается управлением видеоматериалами, а другой – фотографиями. Но сходных черт у этих сайтов больше, чем различий. Оба они управляют цифровыми материалами и дают пользователям возможность делиться друг с другом своими видеороликами и фотографиями. Оба позволяют создавать сети контактов. Пользователи этих сайтов могут создавать свои учетные записи, комментировать материалы других пользователей и отмечать какие-то из них как «избранные» для просмотра в будущем.

Функционально эти сайты очень похожи, но их внешнее представление отличается кардинально. По-разному расположены элементы страниц, отличаются фоновые рисунки, да и вообще они производят совершенно разное впечатление – в каждом из них используется присущее именно ему *оформление* элементов.

Когда мы говорим об *оформлении*, мы имеем в виду уровень представления в системе Drupal. Это место, где разработчик сайта имеет возможность полного управления представлением и может определять, что должно отображаться на странице. С помощью механизма тем оформления в Drupal можно переопределять стили CSS, сценарии JavaScript, изображения и разметку HTML. Этот механизм может также исполь-

зоваться для специального оформления сайта с целью отображения на мобильных устройствах, для переформатирования содержимого с целью отображения в лентах RSS, для отображения аватар рядом с именами пользователей, для полного изменения вывода, получаемого от модуля, и для многого другого. Тема оформления в системе Drupal является настолько широкой, что ей можно посвятить отдельную книгу. Цель этой главы – представить краткий обзор максимально возможного числа концепций, составляющих лишь вершину айсберга, чтобы показать существующие возможности и дать немного самой основной информации о том, как эти возможности использовать. Мы надеемся, что нам удастся рассказать в нашем обзоре достаточно, чтобы идеи, представленные в этой главе, могли послужить для вас отправной точкой, если вы заинтересуетесь вопросом настройки внешнего вида своего сайта на базе Drupal.

Следует заметить: несмотря на то, что в предыдущей части книги нам практически удалось обойтись без фрагментов программного кода, тем не менее, даже для краткого обзора механизма оформления в Drupal потребуется знание основ разметки HTML, стилей CSS и языка PHP. Даже если такая перспектива несколько пугает вас, вы все равно можете бегло просмотреть оставшуюся часть этой главы, так как примеры программного кода, что приводятся здесь, предполагают, что вы знаете хотя бы назначение таких элементов, как `<div>`, `#header` и `foreach`. Если вы достаточно любознательны, но не совсем уверенно чувствуете себя при анализе фрагментов программного кода, вы можете поближе познакомиться с этими тремя технологиями на веб-сайте W3Schools, по адресу *http://w3schools.com*.

Если вы собираетесь следовать за практическими упражнениями этой главы, вам следует установить Drupal, используя профиль установки Drupal по умолчанию. Перейдите на страницу *http://theme.usingdrupal.com*, где вы найдете законченный веб-сайт, а также копию темы, в которой выполнены настройки, описываемые в этой главе.

В центре внимания: механизм поддержки тем оформления

Как уже коротко говорилось в главе 1, модули генерируют содержимое данной страницы, а механизм поддержки тем оформления обеспечивает возможность вмешаться в оформление страницы, прежде чем она будет отображена в броузере. *Тема оформления* – это набор изображений, стилей CSS и (как правило) файлов HTML/PHP, которые изменяют вид страницы по умолчанию. Простым щелчком на радиокнопке вы можете полностью изменить дизайн своего веб-сайта, выбирая другую тему оформления, и можете изменять настройки темы, внося такие коррективы, как подстановка своего логотипа или применение своей цветовой схемы, как было показано в главе 2.

На рис. 11.1 схематически изображен принцип действия механизма поддержки тем оформления. Каждому пути в Drupal соответствует определенный модуль, который отвечает за обработку запроса на получение страницы. Например, путь *node/1* обрабатывается модулем Node, *admin/build/themes* – модулем System, а такой путь, как *albums*, может обрабатываться модулем Views. После того как модуль воспроизведет содержимое страницы, он вызывает систему поддержки тем оформления с помощью специальной функции `theme()`, которая будет подробно рассматриваться ниже, в этой главе.

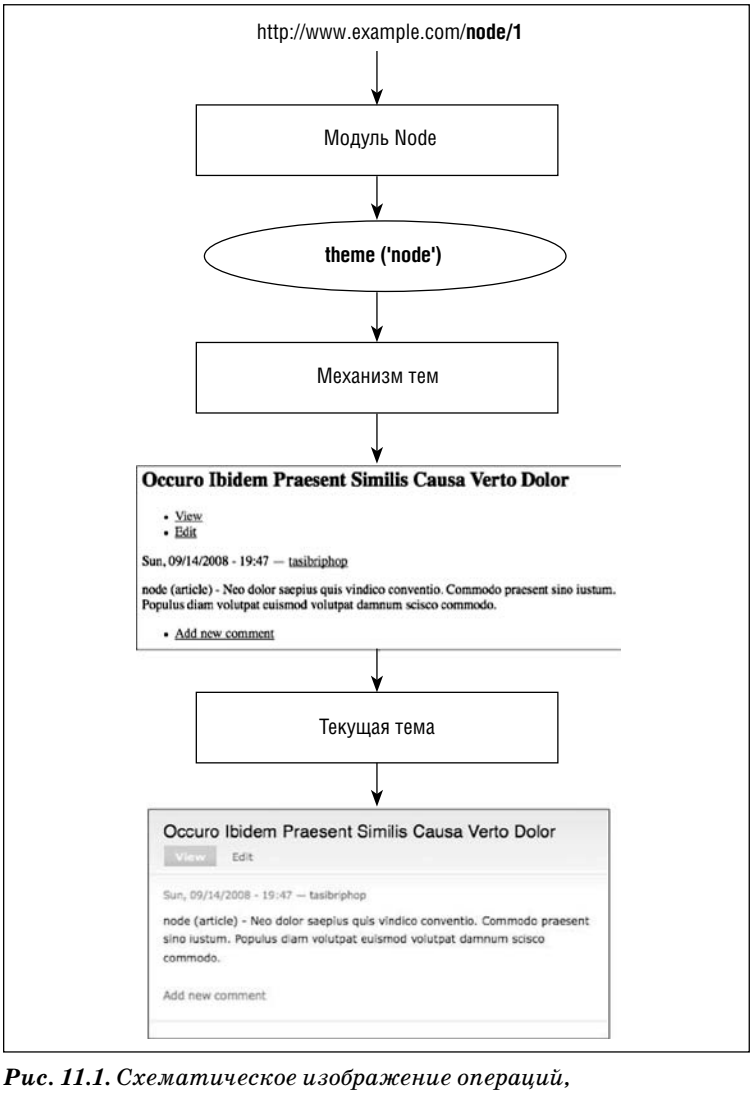


Рис. 11.1. *Схематическое изображение операций, выполняемых Drupal на пути от адреса URL к разметке HTML*

Модификация Drupal

Для чего вообще нужен механизм поддержки тем? Drupal – это же открытый продукт, разве не так? Почему бы просто не отыскать нужное место в файлах ядра Drupal, в котором делается не совсем то, что нам требуется, и не внести в него требуемые изменения?

Для начала заметим, что модификация файлов ядра Drupal – это практически всегда плохая идея. Одна из самых сильных сторон системы Drupal заключается в ее распространенности – все используют одну и ту же отработанную, выверенную и безопасную реализацию. Этот программный код постоянно тестируется и улучшается огромным количеством членов сообщества.

Когда вы вносите изменения в файлы вашей копии Drupal, вы тем самым создаете частную версию Drupal и теряете возможность использовать плоды коллективных усилий. Когда проект Drupal извещает о выходе исправлений в системе безопасности, обновление вашей, частной версии Drupal превращается в задачу трудноразрешимую или вообще невыполнимую. Большинство сайтов, использующих модифицированные файлы Drupal, рано или поздно сталкиваются с невозможностью воспользоваться исправлениями в системе безопасности, улучшениями производительности, новыми возможностями и удобствами в использовании, присущими самой последней версии системы.

Система Drupal спроектирована так, что не требует вносить изменения для ее подгонки под ваши потребности. Она обеспечивает массу возможностей, которые называются *ловушками*, повлиять на содержимое страницы в процессе ее вывода, позволяющих модулям и темам оказывать влияние на базовое функционирование Drupal. Система Drupal всегда будет выполнять действия, предусмотренные по умолчанию, но она дает массу возможностей *переопределить* это поведение по умолчанию. Большинство способов переопределения поведения и представления по умолчанию опирается на простые соглашения об именовании. Если вы дадите своим файлам и функциям соответствующие имена, они обретут особое значение для системы.

Отделив наши настройки, мы легко можем изменить порядок работы системы Drupal без вмешательства в ее программный код. Кроме того, когда придет время обновления сайта, обновление ядра Drupal и сторонних тем пройдет без нашего вмешательства, а нам останется лишь позаботиться об изменениях в своих настройках.

Система поддержки тем оформления в первую очередь состоит из *механизма тем*, который определяет динамические части страницы, а также правила вывода страницы системой Drupal, которые могут определяться и переопределяться текущей *темой оформления*, такой как Garland. Механизм тем способен изменять базовую структуру и разметку содержимого и отобразить целую страницу шаблона с содержимым текущей страницы внутри. Но настоящее волшебство преображения внешнего вида происходит, когда вы сами начинаете определять, как должен выглядеть каждый элемент страницы, в своих собственных темах.

Каждый отдельный элемент страницы, от заголовка в окне броузера до логотипа сайта и до областей, где должны располагаться блоки, и ссылок в полосе меню, проходит через механизм поддержки тем оформления. Степень отличия между сайтом, внешний вид которого явно говорит об использовании системы Drupal, и стильным сайтом зависит от творческого подхода к использованию инструментов, предоставляемых уровнем поддержки тем оформления. Рассмотрим составные элементы тем, с которыми нам предстоит работать, а также обсудим вопросы, как с ними работать.

В этой главе обсуждается механизм тем PHPTemplate (известный также, как механизм шаблонов), который входит в состав ядра Drupal. Однако следует указать, что имеются и другие механизмы тем, включая Smarty и PHPTAL. Полный список поддерживаемых механизмов приводится по адресу *http://drupal.org/project/Theme+Engines*.

Файлы .info

Начиная с версии Drupal 6, темы оформления определяются в виде файлов *theme_name.info*, размещаемых в каталоге с темами, которые часто называют «*файлами .info*». Такой файл определяет различные метаданные для темы, включая название темы, описание, совместимую версию ядра Drupal и какой механизм тем она использует. Помимо этих базовых сведений файл *.info* может также определять области для блоков, стили CSS и используемые файлы JavaScript. Вы можете также определить, какие особенности темы будут доступны администраторам на странице настройки темы (раздел «Настройка тем оформления» в главе 2).

Ниже приводится простой пример, демонстрирующий основные сведения о теме Bluemarine. Полное описание всех атрибутов, доступных в файлах *.info*, можно найти по адресу *http://drupal.org/node/171205*.

```
name = Bluemarine
description = Table-based multi-column theme with a marine and ash color scheme.
core = 6.x
engine = phptemplate
```

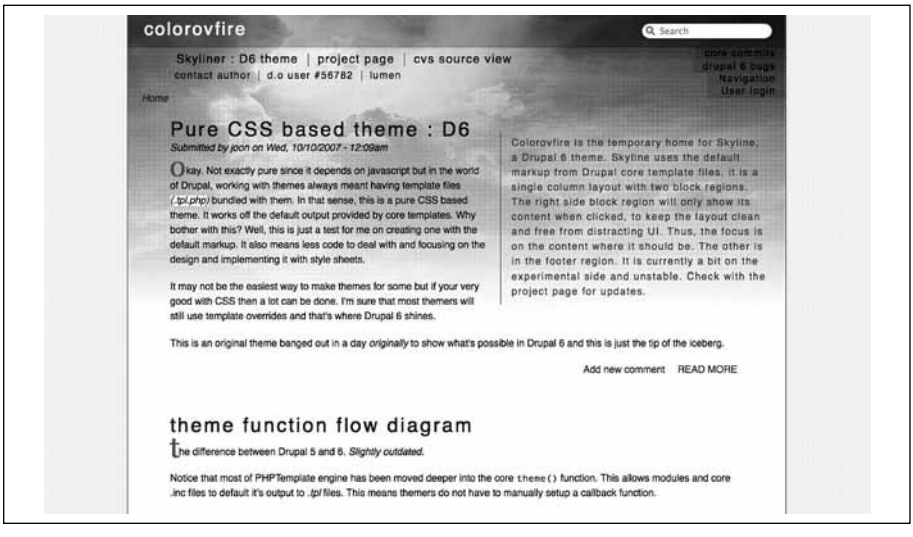


Рис. 11.2. *Тема Skyliner, которая состоит только из стилей CSS, программного кода JavaScript, изображений и файла.info*

Файлы *.info* – это единственные файлы, необходимые для определения темы. Они позволяют создавать темы без использования программного кода PHP или написания своей разметки HTML. Темы могут опираться на разметку HTML по умолчанию, которую воспроизводит система Drupal и в которую при необходимости могут добавляться стили CSS, изображения и сценарии JavaScript. Тема Skyliner (*http://drupal.org/project/skyliner*), изображенная на рис. 11.2, является примером такой темы.

Области

Области (regions) – это участки на страницах сайта, куда вы можете помещать блоки, как мы делали это в главе 2 и в других главах этой книги. По умолчанию Drupal предоставляет пять областей:

- Header (Заголовок)
- Footer (Подвал)
- Left sidebar (Левая колонка)
- Right sidebar (Правая колонка)
- Content (Содержание)

Вы можете изменить этот перечень областей в файле *.info* своей темы. Ниже приводится пример создания новой области Ads (Реклама) и исключения некоторых областей по умолчанию, таких как Header (Заголовок), простым исключением их определений:

```
regions[ads] = Ads
regions[right] = Right sidebar
regions[content] = Content
regions[footer] = Footer
```

Такое определение областей в теме ограничивает возможности администраторов сайта по размещению блоков на странице администрирования. У них не будет возможности использовать область Header (Заголовок) или Left sidebar (Левая колонка). Такая возможность может способствовать созданию специализированных схем размещения элементов страницы, когда нежелательно, чтобы администраторы помещали те или иные элементы в «неподходящее» для этого место.

Существует еще один этап, связанный с отображением содержимого областей для блоков, который заключается в том, чтобы вывести содержимое переменной с именем области в файле *page.tpl.php* темы, например `<?php print $ads; ?>`. Эта команда выведет все блоки, помещенные в указанную область. Мы подробнее поговорим об этом в разделе «Создание новой области».

Если вы определяете область в своем файле *.info*, система Drupal больше не будет использовать свои установки областей по умолчанию. Даже если вы определяете всего одну область, вы должны явно определить все области, которые вы желаете использовать. Например, при добавлении новой области «Ads» (Реклама) вам необходимо также определить области левой колонки, правой колонки, содержания, заголовка и подвала, если вы желаете их использовать.

Свойства

Свойствами (features) называются различные элементы темы, которые могут включаться и выключаться на странице управления темами: Administer (Администрирование)→Site building (Конструкция сайта)→Themes (Темы оформления) (*admin/build/themes*), на вкладке Configure (Настроить). По умолчанию ядро Drupal предоставляет следующие свойства:

- Logo (Логотип) (переменная logo)
- Site name (Название сайта) (переменная name)
- Site slogan (Слоган сайта) (переменная slogan)
- Mission statement (Миссия) (переменная mission)
- User pictures in posts (Аватары в сообщениях) (переменная node_user_picture)
- User pictures in comments (Аватары в комментариях) (переменная comment_user_picture)

- Search box (Строка поиска) (переменная search)
- Shortcut icon (Иконка) (переменная favicon)
- Primary links (Основные ссылки) (переменная primary_links)
- Secondary links (Дополнительные ссылки) (переменная secondary_links)

Если вам потребуется исключить какое-либо свойство, необходимо будет создать собственный список свойств в файле *.info* и закомментировать или удалить нежелательные свойства. Тема Skyliner определяет следующий список свойств:

```
features[] = name
features[] = slogan
features[] = mission
features[] = search
features[] = favicon
features[] = primary_links
features[] = secondary_links
```

Из этого списка исключены свойства Logo (Логотип), User pictures in posts (Аватары в сообщениях) и User pictures in comments (Аватары в комментариях), вследствие этого, например, при использовании этой темы администратор сайта не сможет установить свой логотип.

CSS

В файле *.info* можно определить имена и местоположение всех файлов CSS и JavaScript, используемых темой. Если в файле *.info* отсутствуют дополнительные определения стилей CSS, система Drupal автоматически отыщет и будет использовать файл *style.css*, находящийся в каталоге темы. При желании вы легко можете организовать использование других стилей, помимо *style.css*. Вы также имеете возможность переопределять стили CSS, предоставляемые модулями. Присвоив файлу CSS для темы то же имя, которое присвоено файлу CSS модуля, мы тем самым, сообщим системе Drupal: «Использовать этот файл, а не тот», – и Drupal будет использовать наш файл вместо оригинального. Для примера мы могли бы создать файл *system-menus.css* в своей теме, и он использовался бы вместо файла *system-menus.css*, находящегося в каталоге модуля System (*modules/system*). Ниже приводится пример определения каскадных стилей CSS из файла *.info* темы Skyliner (обратите внимание, что все файлы находятся в каталоге *styles*, внутри корневого каталога темы):

```
stylesheets[all][] = styles/reset.css
stylesheets[all][] = styles/typography.css
stylesheets[all][] = styles/forms.css
stylesheets[all][] = styles/style.css
```

Параметры оптимизации CSS и JavaScript

Система Drupal имеет модульную архитектуру, поэтому даже в случае относительно простого сайта может оказаться так, что к любой заданной странице будет подключаться множество файлов CSS и JavaScript. В версии Drupal 6 в разметку HTML страницы редактирования включается до *семи* тегов подключения файлов CSS и до *шести* тегов подключения файлов JavaScript (и это при отсутствии дополнительных модулей)! И это число быстро растет по мере установки и подключения дополнительных модулей.

Перейдя на страницу Performance (Производительность) Administer (Администрирование)→Site configuration (Настройка сайта)→Performance (Производительность) (*admin/settings/performance*), вы сможете включить оптимизацию CSS и JavaScript. Этот параметр обеспечит объединение всех файлов CSS для данной страницы в один оптимизированный файл, из содержимого которого будут удалены все комментарии и лишние пробелы. Файлы JavaScript обрабатываются аналогичным образом. Вместо 13 файлов, подключаемых изначально, после оптимизации в странице останутся только следующие 4 строки:

```
<link type="text/css" rel="stylesheet" media="all"
           href="/drupal/sites/default/files/css/
           7f66dbf90ba0323c5d322cde426f75ed.css" />
<link type="text/css" rel="stylesheet" media="print"
           href="/drupal/sites/default/files/css/
           bf2acfbc35fa1d13cbc410a9bdc36563.css" />
<script type="text/javascript"
             src="/drupal/sites/default/files/js/
             45caa15f0935ad439814b66bcdf4b022.js">
</script>
<script type="text/javascript">jQuery.extend(Drupal.settings,
  { "basePath": "/drupal/",
    "teaserCheckbox": { "edit-teaser-js": "edit-teaser-include" },
    "teaser": { "edit-teaser-js": "edit-body" } });
</script>
```

Очевидно, что при меньшем количестве файлов требуется меньшее количество обращений к серверу, так как броузер в состоянии определить наличие требуемых файлов в своем кэше. Это замечательная возможность, и мы настоятельно рекомендуем использовать ее для сайтов, работающих с высокой нагрузкой! Только не стоит включать эту возможность во время активной разработки CSS и JavaScript.

JavaScript

Точно так же существует имя файла с программным кодом JavaScript, распознаваемое по умолчанию, – *script.js*, который можно использовать в своих темах. Опять же, у вас есть возможность добавлять дополнительные файлы, указывая их имена в файле *.info*, как можно видеть на примере темы Skyliner:

```
scripts[] = skyliner.js
```

В состав ядра системы Drupal входит библиотека JavaScript jQuery (*http://jquery.com*), благодаря чему вы получаете доступ ко всем возможностям этой библиотеки и отпадает необходимость добавлять свою библиотеку.

Файлы шаблонов

Темы, включающие в себя одни лишь изображения и стили CSS, прекрасно подходят для случаев, когда разметка HTML, генерируемая системой Drupal, удовлетворяет вашим потребностям. Но как быть, если потребуется заключить заголовок узла в дополнительный тег `<div>` или переместить аватару пользователя из верхней части сообщения в нижнюю? В этом вам помогут файлы шаблонов, которые воспроизводят основной объем разметки в системе Drupal.

Комментарии, узлы, блоки и сама страница целиком – все это воспроизводится с помощью *файлов шаблонов*. Имена файлов шаблонов оканчиваются специальным расширением *.tpl.php*. Файлу шаблона присваивается имя, соответствующее элементу, которым он управляет. Например, комментариями управляет файл *comment.tpl.php*, а всей страницей управляет файл *page.tpl.php*.

На рис. 11.3 показана типичная страница на сайте Drupal, отображаемая с применением файлов шаблонов, а на рис. 11.4 изображена другая страница, отображаемая с применением доработанных файлов шаблонов.

Файлы шаблонов в значительной степени состоят из разметки HTML с включениями фрагментов программного кода на языке PHP, отображающими динамические части страницы. Содержимое файла шаблона страницы *page.tpl.php*, в частности, должно быть в основном понятным для вас, если раньше вам приходилось иметь дело с разметкой HTML. Ниже приводится фрагмент файла *page.tpl.php* для темы Zen (*http://drupal.org/project/zen*):

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
lang="<?php print $language->language; ?>"
xml:lang="<?php print $language->language; ?>">
```

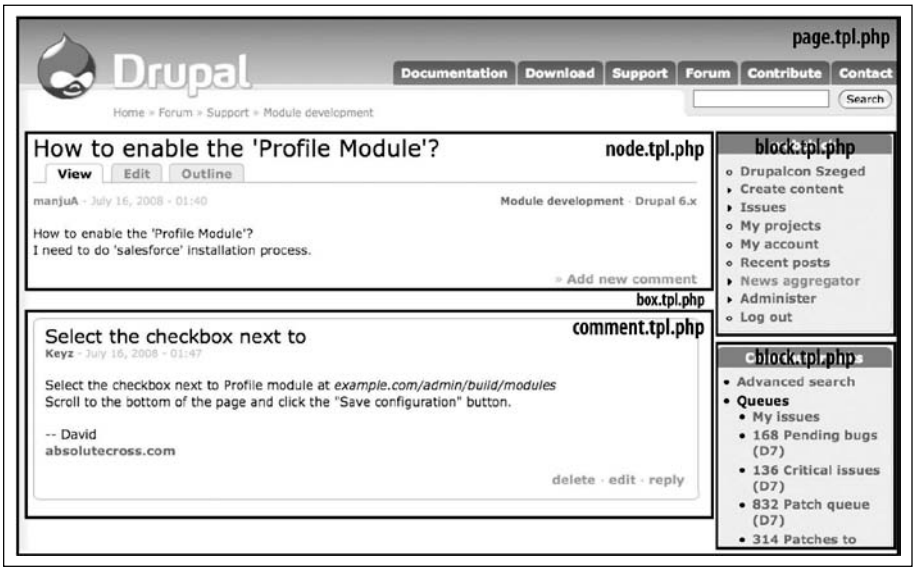


Рис. 11.3. *Пример типичной страницы сайта Drupal.org, сгенерированной с помощью файлов шаблонов*

Рис. 11.4. Сайт SpreadFirefox.com (http://www.spreadfirefox.com), на котором используется более сложная схема размещения элементов, но реализованная с применением тех же самых файлов шаблонов

```
<head>
  <title><?php print $head_title; ?></title>
...
        <div id="content-area">
          <?php print $content; ?>
        </div>
...
  <?php if ($closure_region): ?>
    <div id="closure-blocks" class="region region-closure">
      <?php print $closure_region; ?>
    </div>
  <?php endif; ?>

  <?php print $closure; ?>

</body>
</html>
```

В разметку HTML внедрены фрагменты программного кода PHP, которые предписывают веб-сервеpу: «Вывести содержимое здесь» или «Если в области имеются блоки, поместить их в этот тег `<div>`». Внутри этих фрагментов, заключенных в теги `<?php ?>`, можно заметить слова, начинающиеся со знака доллара (`$`). Эти специальные «слова» на языке PHP называются *переменными*. Переменные представляют динамическое содержимое. Для специалистов, не имеющих опыта работы с языком PHP, это может выглядеть устрашающе, но как только вы познакомитесь с языком PHP достаточно близко, чтобы читать и писать программный код, в этом примере вы неожиданно откроете для себя целую массу удивительных возможностей влиять на поведение системы Drupal.

Файл template.php

К настоящему моменту мы познакомились с основными строительными блоками тем оформления и с некоторыми файлами, из которых состоят темы. Последняя важная часть механизма поддержки тем оформления в Drupal – это возможность полностью изменить вывод, сгенерированный системой Drupal по умолчанию, задействовав свою собственную разметку. Тут начинается самое увлекательное.

Вы можете обнаружить, что, несмотря на огромные возможности, которые предоставляют файлы шаблонов, вы при этом не очень-то можете повлиять на разметку HTML, заключенную в упомянутых выше переменных. Вы можете изменять разметку HTML, окружающую их, как вам заблагорассудится, но как «взломать» сами переменные и изменить их поведение? Все самые тонкие особенности реализации заключены в файле *template.php*. Этот файл дает разработчикам тем оформления возможность по-настоящему продемонстрировать свои умения, позволяя добавлять к темам дополнительные переменные и логику. Они также могут переопределять функции тем оформления – функции со

специальными наименованиями, которые используются модулями для отображения разметки HTML без использования файла *.tpl.php*.

Файл *template.php* находится за пределами «царства HTML» и является обычным файлом с программным кодом на языке PHP, в котором используются функции. Этот файл может выглядеть устрашающе для тех, кто не знаком с PHP, но если вы потратите некоторое время на изучение PHP до уровня, который позволит использовать этот файл, вы получите в свое распоряжение небывалую мощь в управлении своими темами оформления. Ниже приводится небольшой фрагмент из файла *template.php* для темы Foundation (*http://drupal.org/project/foundation*), в котором создается новая переменная `$authored`, используемая в файле *node.tpl.php*, и изменяется форма поиска по умолчанию, из которой удаляется заголовок:

```
<?php
// $Id: template.php,v 1.3 2008/06/23 12:08:02 add1sun Exp $

/**
 * Переопределение или добавление переменных PHPTemplate в шаблоны узла.
 */
function foundation_preprocess_node(&$vars) {
  // Отделить информацию об авторе от переменной $submitted.
  $vars['authored'] = t('Submitted by') .' '. $vars['name'];
}

/**
 * Переопределить форму поиска (в теме, не в блоке), чтобы убрать заголовок.
 */
function foundation_search_theme_form($form) {
  unset($form['search_theme_form']['#title']);
  return drupal_render($form);
}
```

Обратите внимание, что файл *template.php* имеет открывающий тег PHP (`<?php`), но не имеет закрывающего тега (`?>`). Это стандарт записи программного кода, принятый в сообществе Drupal. Наличие закрывающего тега иногда может приводить к ошибкам, поэтому оптимальным решением считается удаление этого тега из файлов. Подробнее об этом стандарте можно прочитать на странице *http://drupal.org/node/545*.

Всякий раз, когда вы что-то переопределяете в теме оформления, вы должны сообщить об этом системе Drupal. Система Drupal стремится оптимизировать свою производительность и поэтому сохраняет в кэше массу информации, обеспечивая тем самым более высокую скорость работы. Когда вы изменяете файлы, которые могут сохраняться в кэше, вам необходимо заставить Drupal обновить их.

В первую очередь при работе над темой оформления следует отключить любое кэширование на сайте. Сделать это можно на странице Administer (Администрирование)→Site configuration (Настройка сайта)→Performance (Производительность) (*admin/settings/performance*), выключив различные параметры оптимизации и кэширования. Кроме того, при переопределении каких-либо элементов необходимо очистить кэш реестра темы, щелкнув на кнопке Clear cached data (Очистить кэш данных), которая находится внизу этой страницы.

Помимо этого модуль Devel (*http://drupal.org/project/devel*) предоставляет удобный блок со ссылкой Clear cache (Очистить кэш), что немного упрощает выполнение этой операции.

Инструменты разработчика тем оформления

Прежде чем перейти к разделу «Практика: создание собственной темы», мы должны отвлечься на несколько минут, чтобы коротко рассмотреть некоторые инструменты, которые многие разработчики считают полезными для организации эффективной работы:

Текстовый редактор

Для работы с файлами в системе Drupal вам потребуется текстовый редактор. Текстовые редакторы – это редакторы обычного текста, которые не добавляют никакой дополнительной разметки, например, для оформления фрагментов текста жирным или курсивным шрифтом. Не используйте текстовые процессоры, такие как Microsoft Word. Вместо них используйте такие текстовые редакторы, как Notepad++ (*http://notepad-plus.sourceforge.net*) для Windows, TextWrangler (*http://www.barebones.com/products/textwrangler*) или Smultron (*http://smultron.sourceforge.net*) для Mac и Kate (*http://kate-editor.org*) или gedit (*http://www.gnome.org/projects/gedit*) для Linux. Удобно пользоваться редакторами, которые поддерживают такие возможности, как выделение синтаксиса и нумерация строк. За дополнительной информацией о различных редакторах обращайтесь по адресу *http://drupal.org/node/147789*.

Firebug (https://addons.mozilla.org/en-US/firefox/addon/1843)

Данный инструмент – лучший друг разработчика тем оформления. Это расширение для броузераFirefox, позволяющее выбирать элементы страницы в окне броузера и сразу же получать определения стилей CSS, оказывающих влияние на их оформление, а также редактировать стили CSS и сразу же наблюдать результаты. Кроме того, этот инструмент отлично подходит для работы с программным кодом JavaScript и позволяет проверять идеи, не покидая окна броузера.

Web Developer Toolbar (https://addons.mozilla.org/en-US/firefox/addon/60)

Это еще одно расширение для броузера Firefox. Оно содержит множество полезных инструментов, позволяющих быстро изменять размеры окна броузера, отображать информацию о различных элементах страницы, производить различные проверки (CSS, HTML, доступности и так далее), изменять и опробовать формы.

Модуль Devel (http://drupal.org/project/devel)

В действительности модуль Devel является пакетом модулей и представляет собой настоящий комплект инструментов разработчика (и разработчика тем оформления). Из наиболее полезных инструментов для разработчика тем можно упомянуть блок, содержащий удобную ссылку Clear cache (Очистить кэш), и модуль Devel Generate, который позволит вам очень быстро заполнить сайт фиктивными пользователями и материалами. Он поможет вам увидеть, как ваша тема взаимодействует с содержимым, даже если сайт еще не имеет реального информационного наполнения.

Модуль Theme Developer (входит в состав пакета Devel)

Этот модуль входит в состав пакета Devel, но он обладает настолько впечатляющими возможностями, что заслуживает отдельного упоминания. Включив модуль Theme Developer, вы сможете увидеть, какие элементы страницы прошли обработку функциями темы или шаблона, просто щелкая на них и получая информацию о том, как эти элементы были сгенерированы и какие переменные им доступны.

Ниже в этой главе мы рассмотрим некоторые простые примеры переопределения вывода разметки HTML с помощью файла *template.php*.

Практика: создание собственной темы

Посмотрим, как можно создать собственную тему оформления сайта на базе Drupal, изменив одну из существующих. Имеется множество неплохих тем, которые могут служить «основой» для дальнейшего развития. В репозитории тем (*http://drupal.org/project/Themes*), созданных сторонними разработчиками, имеется несколько тем, специально предназначенных для этого. Одной из наиболее часто рекомендуемых является тема Zen (*http://drupal.org/project/zen*). В нашем примере мы возьмем за основу простую тему, входящую в состав ядра: Bluemarine, изображенную на рис. 11.5.

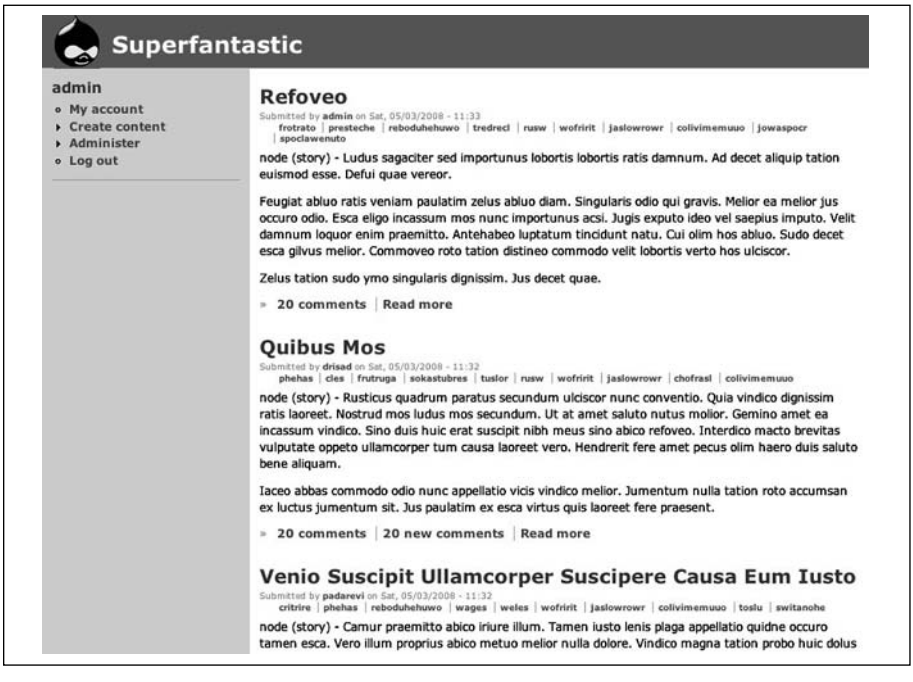


Рис. 11.5. *Тема Bluemarine, входящая в состав ядра Drupal*

Тема Bluemarine использовалась как тема по умолчанию до версии Drupal 5. В ней применялась табличная верстка, что вообще считается плохой практикой в наши дни, когда широко стали использоваться приемы верстки на основе стилей CSS. Однако в рамках нашей задачи эту тему вполне можно использовать в качестве простой и понятной отправной точки. Посмотрим, как можно настроить ее.

На рис. 11.6 приводится содержимое каталога, где находятся файлы темы Bluemarine. В каталоге присутствуют файлы шаблонов для блоков, контейнеров, комментариев, узлов (всех типов) и страниц. Здесь также имеется файл с изображением логотипа по умолчанию и снимок с экрана, с изображением страницы со списком. Обязательный файл *.info* также присутствует здесь. В каталоге имеется основной файл с таблицами стилей для темы, с именем *style.css*, и еще один файл, *style-rtl.css*, который используется системой Drupal для вывода информации на языках с письмом справа налево, таких как арабский или иврит.

Создание копии темы

Прежде чем приступать к изменению темы, необходимо создать ее копию и поместить ее в в соответствующий подкаталог в каталоге *sites*. Каталог *sites* можно представлять себе как «нечто, принадлежащее только вашему сайту». Все настройки, устанавливаемые модули и темы оформ-

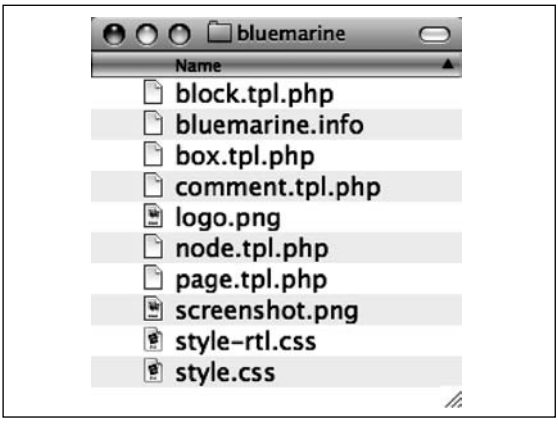


Рис. 11.6. *Файлы темы оформления Bluemarine*

ления должны помещаться в этот каталог. В большинстве случаев темы помещаются в каталог *sites/all/themes*. Система Drupal пытается искать темы во всех вложенных подкаталогах, поэтому вы свободно можете определять любую организацию подкаталогов по своему усмотрению.

1. Проверьте наличие каталога *sites/all/themes* и скопируйте туда каталог *bluemarine*.
2. Переименуйте новый каталог из *bluemarine* в *newmarine*.
3. Переименуйте файл *bluemarine.info* в *newmarine.info*.
4. Откройте файл *newmarine.info* в текстовом редакторе и измените его так, чтобы он отражал новое название темы. Эта информация будет отображаться на странице со списком тем в разделе администрирования Drupal:

```
; $Id$
name = Newmarine
description = Table-based multi-column theme based on Bluemarine.
version = VERSION
core = 6.x
engine = phptemplate
```

Несложно заметить, что файлы, загружаемые с сайта Drupal.org, всегда начинаются с комментария, содержащего текст `$Id:`, название и некоторую дополнительную информацию. Это специальный тег ID системы управления версиями (CVS), используемой проектом Drupal.org для управления всем программным кодом ядра и сторонних модулей. Ваши собственные файлы необязательно должны содержать эти строки, но их наличие никак не помешает. Если вы когда-нибудь решите передать результаты своего труда сообществу, вам придется добавить эти строки в начало каждого файла. В самом простом случае достаточно указать единственную строку `$Id$`.

5. Перейдите на страницу управления темами: Administer (Администрирование)→Site building (Конструкция сайта)→Themes (Темы оформления) (*admin/build/themes*) и включите новую тему, щелкнув на радиокнопке Default (По умолчанию) рядом с ее названием, как показано на рис. 11.7. Щелкните на кнопке Save configuration (Сохранить конфигурацию), чтобы переключиться на использование новой темы.

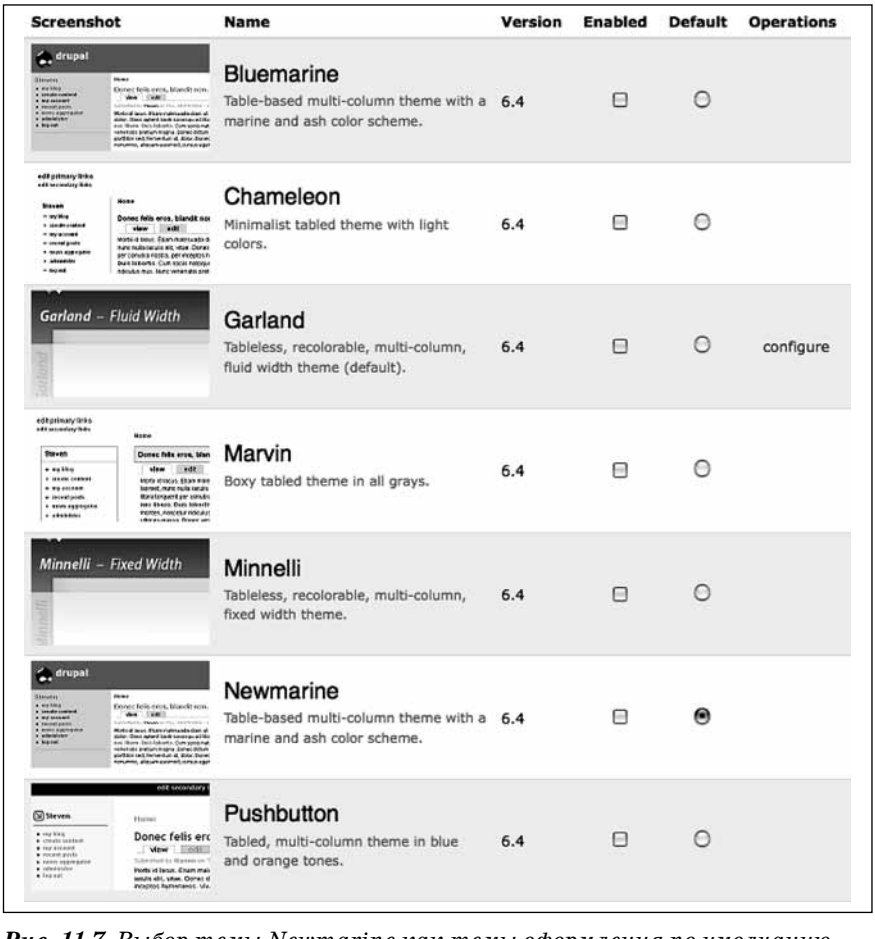


Рис. 11.7. *Выбор темы Newmarine как темы оформления по умолчанию*

Теперь вы должны увидеть свой сайт в новом оформлении. Сейчас он выглядит так же, как и при использовании темы Bluemarine, но через несколько минут мы изменим его.

Последнее, что необходимо сделать, прежде чем приступать к изменению темы, – это наполнить сайт некоторым содержимым, чтобы можно было наблюдать, как изменяется его оформление:

1. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Modules (Модули) (*admin/build/modules*) и включите следующий модуль:
   - Development (Разработка)
     – Devel generate
2. Перейдите на страницу Administer (Администрирование)→Generate items (Создание элементов) (*admin/generate*) и щелкните на ссылке Generate categories (Создать категории). Мы должны сначала создать категории, чтобы потом можно было назначать их узлам, которые мы создадим на следующем шаге.
3. Значения по умолчанию вполне пригодны для наших целей, поэтому просто щелкните на кнопке Do it! (Выполнить), чтобы создать категории.
4. Вернитесь на страницу Administer (Администрирование)→Generate items (Создание элементов) (*admin/generate*) и щелкните на ссылке Generate content (Создать содержимое). Здесь также большая часть значений по умолчанию вполне подходит для нашего примера, но, тем не менее, нам потребуется кое-что добавить к содержимому.
5. В текстовом поле How many comments per node would you like to generate? (Количество комментариев, которые необходимо сгенерировать для каждого узла) введите значение 100, чтобы мы могли также видеть, как будут выглядеть комментарии на нашем сайте.
6. Нам также потребуются рубрики таксономии, поэтому отметьте флажок Add taxonomy terms to each node (Добавить термины таксономии в каждый узел).
7. Щелкните на кнопке Do it! (Выполнить), и вы получите заполненный сайт, готовый к дальнейшей работе.

Изменение CSS

Прежде чем приступать к изменению стилей CSS, перейдите на страницу Administer (Администрирование)→Site configuration (Настройка сайта)→Performance (Производительность) (*admin/settings/performance*) и установите параметр Optimize CSS files (Оптимизировать CSS-файлы) в значение Disabled (Отключено). Если этот параметр оставить включенным, внесение изменений в файлы CSS темы не будет сразу же отражаться на оформлении сайта.

Теперь можно приступать к изменению темы. Для начала добавим рамки вокруг элементов содержимого, чтобы обеспечить их визуальное выделение на странице:

1. Откройте файл *style.css* темы Newmarine в текстовом редакторе.
2. Отыщите следующее правило (на строке 232):

```
.node {
  margin: .5em 0 2em; /* LTR */
}
```

3. Измените его, как показано ниже:

```
.node {
  margin: .5em 0 2em; /* LTR */
  border: 1px dashed #ccc;
  padding: .5em;
}
```

4. Сохраните файл и обновите главную страницу веб-сайта.

Теперь все узлы на странице должны быть окружены пунктирными рамками, как показано на рис. 11.8. Если этого не произошло, попробуйте принудительно обновить страницу в броузере, для чего, нажав и удерживая клавишу Shift, щелкните на кнопке Reload (Обновить), расположенной на панели инструментов броузера.

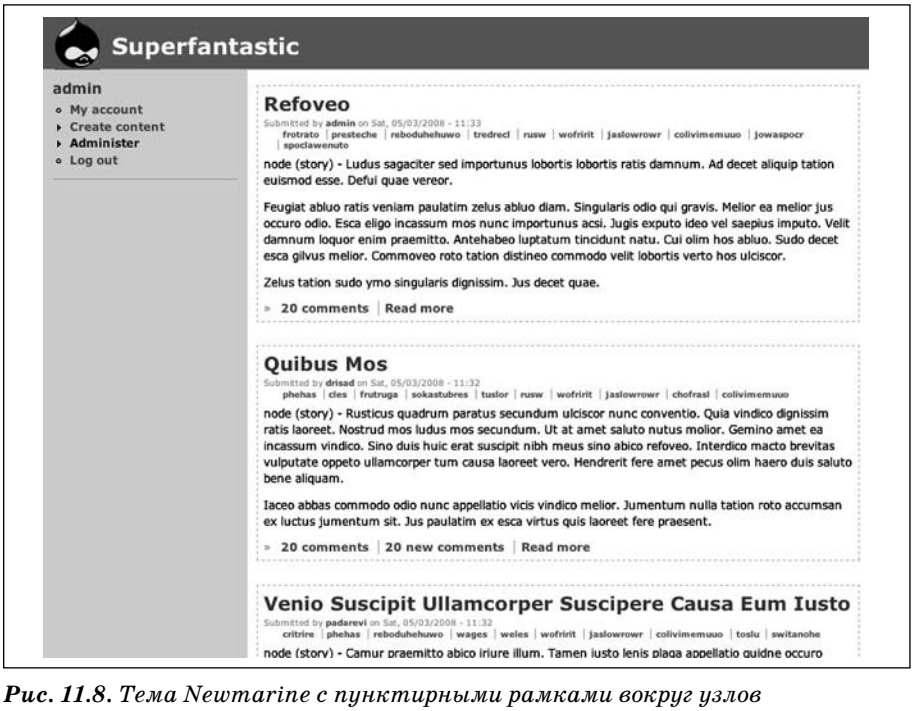


Рис. 11.8. *Тема Newmarine с пунктирными рамками вокруг узлов*

Итак, теперь у нас имеется новая тема, некоторое фиктивное информационное наполнение сайта, с которым можно работать, и несколько изменений в стилях CSS. Следующее, что мы сделаем, – внесем более существенные изменения в разметку HTML.

В центре внимания: файлы шаблонов

Самое интересное, что разметку HTML достаточно просто можно переопределить с помощью файлов шаблонов. Все, что для этого необходимо сделать, – отыскать файл шаблона по умолчанию, скопировать его в папку с темой и отредактировать по своему усмотрению. Многие темы, входящие в состав ядра Drupal, по умолчанию используют следующие файлы шаблонов:

page.tpl.php (находится в каталоге modules/system)

Определяет разметку целой страницы. Здесь подразумевается не тип содержимого Page (Страница), а все окно броузера, в котором отображаются ваши материалы и блоки.

node.tpl.php (located in modules/node)

Используется для отображения отдельных узлов.

block.tpl.php (находится в каталоге modules/system)

Используется для отображения всех блоков сайта.

comment.tpl.php (находится в каталоге modules/comment)

Используется для отображения отдельных комментариев.

box.tpl.php (находится в каталоге modules/system)

Это редко применяемый устаревший шаблон. Используется ядром Drupal для отображения контейнерных элементов, вмещающих результаты поиска и списки комментариев, присоединенных к узлу.

Помимо этих файлов шаблонов узлов и комментариев в папках модулей, созданных этими модулями, могут присутствовать также другие шаблоны. Это справедливо и для модулей ядра, и для сторонних модулей. Например, шаблоны форумов можно найти в папке модуля Forum (*modules/forum*), а шаблоны календарей – в папке модуля Calendar (*sites/all/modules/calendar*).

В дополнение к основным файлам шаблонов, входящим в состав ядра, имеется ряд дополнительных «рекомендуемых» шаблонов – например, *page-front.tpl.php*, который воздействует только на разметку главной страницы сайта. Имеется также возможность управлять оформлением типов содержимого по отдельности, с помощью рекомендуемых шаблонов. Например, можно настроить сообщения в блогах так, что аватара автора будет отображаться внизу, а заметки и страницы вообще не будут содержать аватары. Файлам шаблонов, применяемым для отображения определенных типов содержимого, даются имена в соответствии со следующей схемой именования: *node-nodetype.tpl.php*, где *nodetype* – это, например, «story», «page», «forum», «blog» или какое-либо другое название типа содержимого. Эти анализируемые системой Drupal имена можно найти в колонке Type (Тип) в списке Content types (Типы материалов) на странице Administer (Администрирование)→Content management

(Содержание)→Content types (Типы материалов) (*admin/content/types*), что показано на рис. 11.9.

Список имен рекомендуемых шаблонов и шаблонов ядра, которые можно использовать, приводится в справочном руководстве, на странице *http://drupal.org/node/190815*.

Name	Type	Description	Operations			
Book page	book	A *book page* is a page of content, organized into a collection of related entries collectively known as a *book*. A *book page* automatically displays links to adjacent pages, providing a simple navigation system for organizing and reviewing structured content.	edit	add field	manage fields	delete
Page	page	A *page*, similar in form to a *story*, is a simple method for creating and displaying information that rarely changes, such as an "About us" section of a website. By default, a *page* entry does not allow visitor comments and is not featured on the site's initial home page.	edit	add field	manage fields	delete
Story	story	A *story*, similar in form to a *page*, is ideal for creating and displaying content that informs or engages website visitors. Press releases, site announcements, and informal blog-like entries may all be created with a *story* entry. By default, a *story* entry is automatically featured on the site's initial home page, and provides the ability to post comments.	edit	add field	manage fields	delete

Рис. 11.9. *Названия анализируемых системой типов содержимого, которые могут использоваться для именования файлов шаблонов*

Практика: работа с файлами шаблонов

Модификация файла шаблона

Теперь рассмотрим порядок изменения файла *page.tpl.php*. Этот файл определяет общую структуру разметки HTML для целой страницы. Механизм поддержки тем оформления и активные модули определяют ряд переменных, доступных в этом файле, содержимое которых просто выводится инструкциями `print` языка PHP. Давайте изменим структуру темы, переместив навигационную цепочку в область заголовка страницы. Нам необходимо, чтобы навигационная цепочка выглядела, как показано на рис. 11.10.

Внутри шаблонов доступно множество переменных, но не все они используются в темах, входящих в состав ядра Drupal. Список переменных можно найти по адресу *http://drupal.org/phptemplate*, где переменные для каждого шаблона перечисляются в соответствующем разделе. Однако модули могут дополнять и изменять этот список, поэтому самый лучший способ познакомиться со списком доступных переменных – это задействовать модуль Devel Themer, который мы обсудим чуть ниже.

1. В своей работе мы будем использовать модуль Theme developer, поэтому перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Modules (Модули) (*admin/build/modules*) и включите следующие модули:

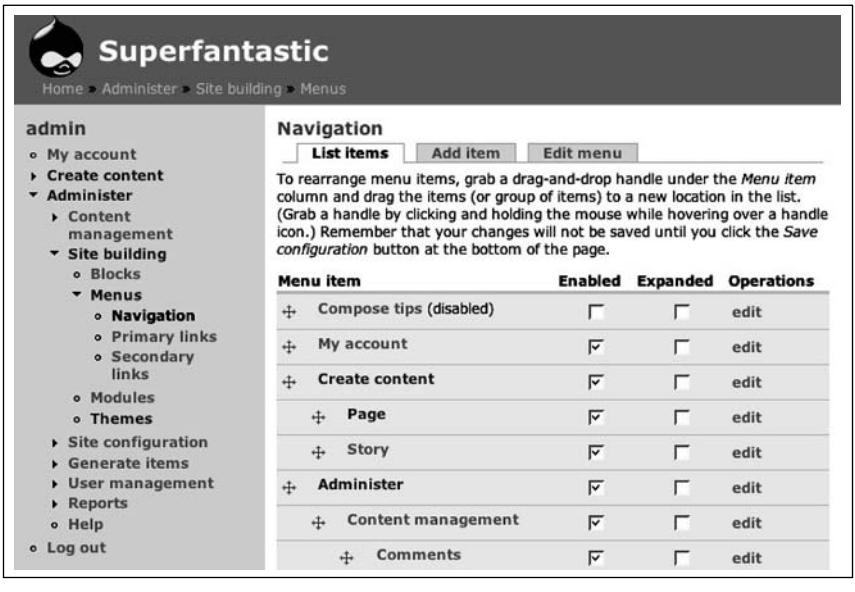


Рис. 11.10. *Навигационная цепочка, отображаемая в заголовке*

- Development (Разработка)
  - Devel
  - Theme developer

2. Вы по-прежнему должны остаться на странице управления модулями, где в верхней части присутствует навигационная цепочка. Обратите также внимание на полупрозрачный блок, появившийся в нижнем левом углу окна броузера с флажком Themer info (Информация о теме), как показано на рис. 11.11. Отметьте флажок, чтобы активировать модуль Theme developer для страницы.

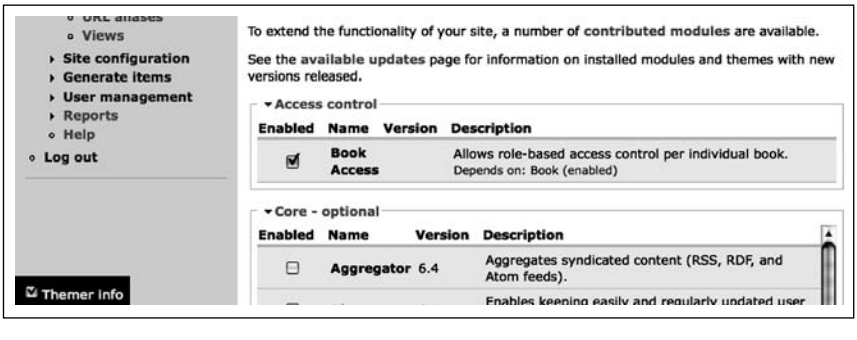
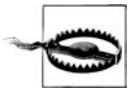


Рис. 11.11. *После включения модуля Theme developer появляется флажок Themer info (Информация о теме)*



Если инструмент разработчика тем не работает, это может быть обусловлено тем, что интерпретатор PHP исчерпал доступную ему память. Попробуйте открыть какую-нибудь другую страницу, отличную от страницы управления модулями, для отображения которой требуется большой объем памяти, тоже связанную с инструментом разработчика тем (который сам по себе потребляет значительный объем памяти). Следует отключать инструмент разработчика тем, когда он не используется.

3. После установки флажка в верхнем правом углу окна броузера появится новый серый блок с заголовком Drupal Themer Information (Информация для разработчика тем). Следуя подсказке, которая выводится в этом блоке: «Click on any element to see information about the Drupal theme function or template that created it» (Щелкните на любом элементе, чтобы увидеть информацию о функции в теме Drupal или о шаблоне, создавшем его), наведите указатель мыши на навигационную цепочку. Вы должны увидеть, как вокруг нее появится красная рамка. После появления красной рамки вокруг выбранного элемента щелкните на странице, чтобы получить информацию о нем. Красная рамка окрасится в серый цвет, и вы должны увидеть всплывающее окно, как показано на рис. 11.12.
4. Искомой информацией в данном случае является имя файла шаблона, который выводит данный элемент на странице. Если взглянуть на первый раздел в окне Drupal Themer Information (Информация для разработчика тем), можно увидеть строку Parents: page.tpl.php (Родители: page.tpl.php). То есть наша навигационная цепочка выводится шаблоном *page.tpl.php*.

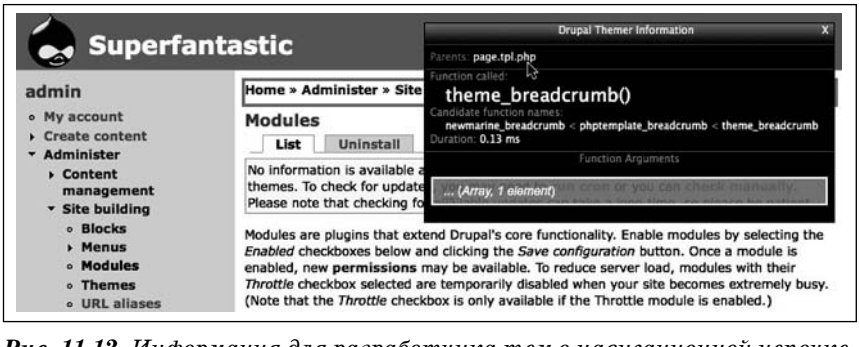


Рис. 11.12. *Информация для разработчика тем о навигационной цепочке*

5. Откройте файл *page.tpl.php*, находящийся в папке темы Newmarine, и переместитесь к строке 29, где производится вывод переменной `$header`. Ниже, в строке 42, выполняется операция вывода переменной `$breadcrumb`, которую нам необходимо переместить:

```
29  <tr>
30    <td colspan="2"><div><?php print $header ?></div></td>
31  </tr>
32</table>
33
34<table border="0" cellpadding="0" cellspacing="0" id="content">
35  <tr>
36    <?php if ($left) { ?><td id="sidebar-left">
37      <?php print $left ?>
38    </td><?php } ?>
39    <td valign="top">
40      <?php if ($mission) { ?><div id="mission"><?php print $mission ?>
        </div><?php } ?>
41    <div id="main">
42      <?php print $breadcrumb ?>
43      <h1 class="title"><?php print $title ?></h1>
```

6. Теперь изменим файл так, чтобы навигационная цепочка выводилась непосредственно за заголовком, удалив строку 42 и переместив ее содержимое в строку 30, как показано ниже:

```
29  <tr>
30    <td colspan="2"><div><?php print $header ?><?php print $breadcrumb ?>
      </div></td>
31  </tr>
32</table>
33
34<table border="0" cellpadding="0" cellspacing="0" id="content">
35  <tr>
36    <?php if ($left) { ?><td id="sidebar-left">
37      <?php print $left ?>
38    </td><?php } ?>
39    <td valign="top">
40      <?php if ($mission) { ?><div id="mission"><?php print $mission ?>
        </div><?php } ?>
41      <div id="main">
42
43        <h1 class="title"><?php print $title ?></h1>
```

7. По окончании сохраните файл.
8. Перезагрузите страницу, после чего в области заголовка страницы должна появиться навигационная цепочка. Сейчас синяя цепочка отображается на синем фоне, поэтому необходимо немного изменить стили CSS, чтобы текст цепочки не сливался с фоном заголовка.
9. Отыщите следующее правило в строке 171, в файле *style.css*:

```
.breadcrumb {
  margin-bottom: .5em;
}
```

10. И замените его следующими правилами:

```
.breadcrumb {
  margin-left: 2em;
}
.breadcrumb a {
  color: #ccc;
  font-weight: normal;
}
```

Теперь наша навигационная цепочка выводится в заголовке страницы и оформлена так, что уже не сливается с фоном.

Оформление отдельных типов содержимого

Чтобы показать, как действуют шаблоны, управляющие отображением отдельных типов содержимого, мы создадим файл шаблона, определяющий оформление узлов типа Story (Статья). Нам необходимо переместить информацию об авторе заметки и список терминов таксономии (рубрик) в нижнюю часть элемента содержимого (рис. 11.13).

Рис. 11.13. *Информация об авторе и термины таксономии перемещены в нижнюю часть элемента содержимого*

1. Скопируйте файл *node.tpl.php* из каталога *newmarine* в новый файл с именем *node-story.tpl.php*.
2. Откройте файл *node-story.tpl.php* в текстовом редакторе. Он должен выглядеть, как показано ниже:

```
<?php
// $Id: node.tpl.php,v 1.7 2007/08/07 08:39:36 goba Exp $
?>
<div class="node<?php if ($sticky) { print " sticky"; } ?><?php if
(!$status)
{ print " node-unpublished"; } ?>">
<?php if ($picture) {
      print $picture;
}?>
<?php if ($page == 0) { ?><h2 class="title"><a href="<?php print $node_url?>">
<?php print $title?></a></h2><?php }; ?>
<span class="submitted"><?php print $submitted?></span>
<div class="taxonomy"><?php print $terms?></div>
<div class="content"><?php print $content?></div>
<?php if ($links) { ?><div class="links">&raquo;
<?php print $links?></div><?php }; ?>
</div>
```

3. Теперь переместим вывод информации об авторе и терминов таксономии после вывода основного содержимого узла:

```
<?php
// $Id: node.tpl.php,v 1.7 2007/08/07 08:39:36 goba Exp $
?>
<div class="node<?php if ($sticky) { print " sticky"; } ?><?php if
(!$status)
{ print " node-unpublished"; } ?>">
<?php if ($picture) {
      print $picture;
}?>
<?php if ($page == 0) { ?><h2 class="title"><a href="<?php print $node_url?>">
<?php print $title?></a></h2><?php }; ?>
<div class="content"><?php print $content?></div>
<span class="submitted"><?php print $submitted?></span>
<div class="taxonomy"><?php print $terms?></div>
<?php if ($links) { ?><div class="links">&raquo;
<?php print $links?></div><?php }; ?>
</div>
```

Точно таким же способом мы можем изменять порядок вывода этих элементов на странице, используемый системой Drupal по умолчанию.

Переопределение файла шаблона модуля

Теперь продемонстрируем, как переопределять файлы шаблонов модуля на примере файла *comment-wrapper.tpl.php* модуля Comment. Этот

файл просто обертывает комментарии дополнительным элементом `div`. Мы добавим в этот файл вывод текста «Comments:» (Комментарии) перед списком комментариев к узлам страницы. Наша цель – добиться отображения комментариев, как показано на рис. 11.14.

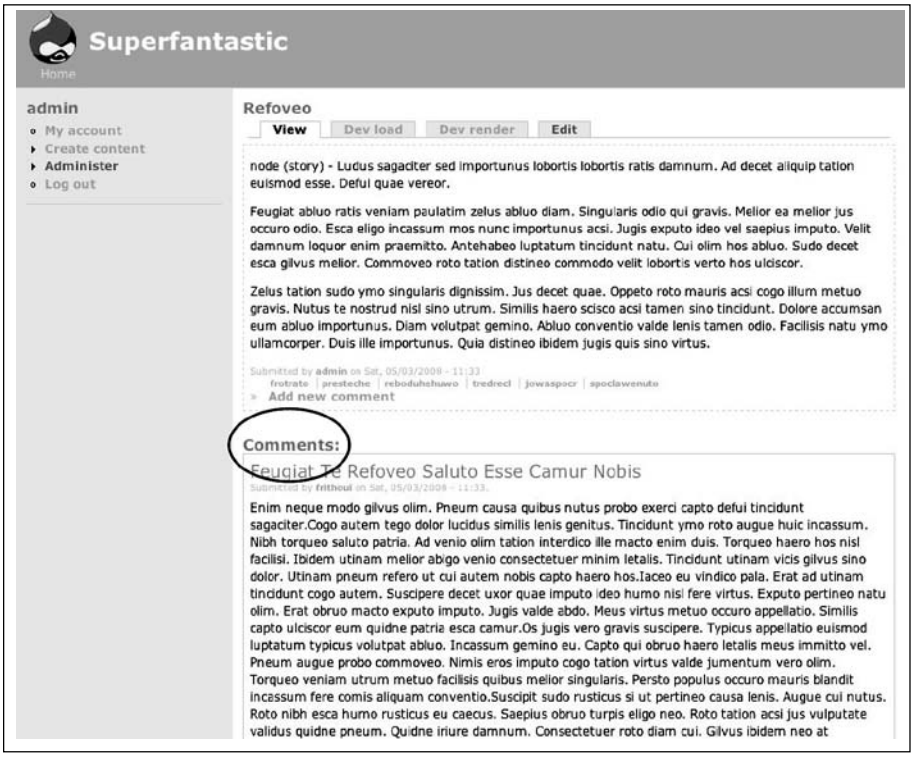


Рис. 11.14. *Список комментариев начинается с заголовка «Comments:» (Комментарии)*

Для этого выполните следующие действия:

1. Отыщите файл *modules/comment/comment-wrapper.tpl.php* и скопируйте его в каталог темы *sites/all/themes/custom/newmarine*. Обратите внимание, что файл должен копироваться, а не перемещаться, потому что оригинальный файл еще может потребоваться в случае перехода к использованию другой темы оформления.
2. Откройте скопированный файл в текстовом редакторе. Он содержит большое число комментариев, описывающих доступные переменные:

```
<?php
// $Id: comment-wrapper.tpl.php,v 1.2 2007/08/07 08:39:35 goba Exp $
/**
 * @file comment-wrapper.tpl.php
 * Реализация темы по умолчанию для оформления комментариев.
```

```
 *
 * Доступные переменные:
 * - $content: Все комментарии на данной странице. Также содержит
 *   элементы управления сортировкой и формы ввода комментариев,
 *   если сайт предусматривает такую возможность.
 *
 * Следующие переменные могут использоваться
 * для получения информации о контексте.
 * - $node: Узел, к которому присоединен комментарий.
 * Ниже приводятся переменные и их допустимые значения,
 * которые должны использоваться в операциях сравнения.
 * - $display_mode
 *   - COMMENT_MODE_FLAT_COLLAPSED
 *   - COMMENT_MODE_FLAT_EXPANDED
 *   - COMMENT_MODE_THREADED_COLLAPSED
 *   - COMMENT_MODE_THREADED_EXPANDED
 * - $display_order
 *   - COMMENT_ORDER_NEWEST_FIRST
 *   - COMMENT_ORDER_OLDEST_FIRST
 * - $comment_controls_state
 *   - COMMENT_CONTROLS_ABOVE
 *   - COMMENT_CONTROLS_BELOW
 *   - COMMENT_CONTROLS_ABOVE_BELOW
 *   - COMMENT_CONTROLS_HIDDEN
 *
 * @see template_preprocess_comment_wrapper()
 * @see theme_comment_wrapper()
 */
?>
<div id="comments">
<?php print $content; ?>
</div>
```

3. Добавьте новый элемент `<h2>`, содержащий текст `Comments:`. В нашем примере измененного файла мы удалили комментарии, чтобы сократить его, но их можно было и оставить. Кроме того, мы обернули текст функцией `t()`, чтобы в случае необходимости он был доступен механизму перевода Drupal:

```
<div id="comments">
<h2 class="title"><?php print t('Comments:'); ?></h2>
<?php print $content; ?>
</div>
```

4. Не забывайте – чтобы система Drupal обнаружила новый шаблон, необходимо обновить реестр тем. Для этого перейдите на страницу Administer (Администрирование)→Site configuration (Настройка сайта)→Performance (Производительность) (*admin/settings/performance*) и щелкните на кнопке Clear cached data (Очистить кэш данных), как показано на рис. 11.15.

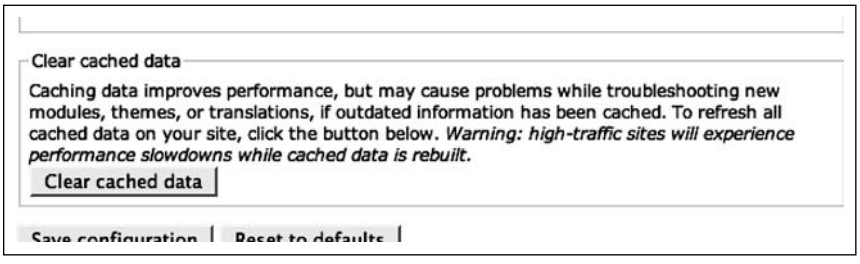


Рис. 11.15. *Кнопка Clear cached data (Очистить кэш данных), которую следует использовать, когда не происходит немедленного изменения оформления*

Создание новой области

Как мы уже говорили, система Drupal предоставляет несколько областей по умолчанию. В специальной области «content» (содержание) блоки проявляют не совсем обычное поведение: при помещении блока в эту область, вместо того чтобы расположиться над обычным содержимым, он фактически располагается *под* ним. По умолчанию нет никакого способа добавить блок в верхнюю часть области содержания. Однако для решения подобных проблем мы обладаем мощью файла *.info* и файлов шаблонов, поэтому давайте добавим новую область, которая позволит нам помещать блоки в требуемое нам место. На рис. 11.16 показан сайт с областью, включающей блок Powered by Drupal (Под управлением Drupal) и помещенной над содержанием.

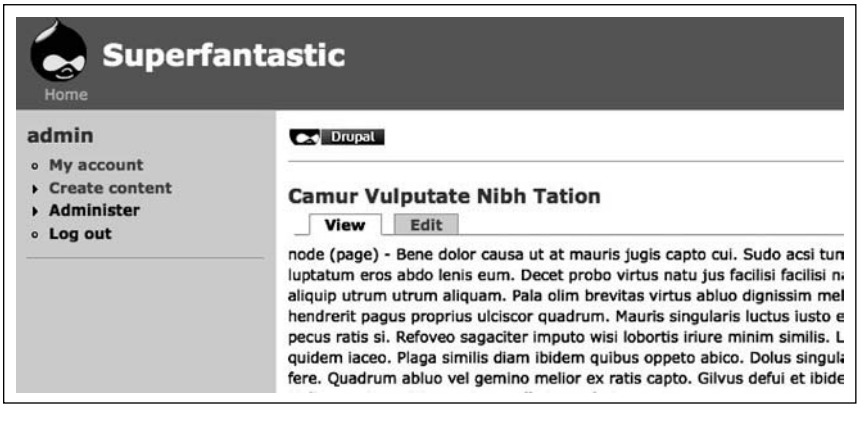


Рис. 11.16. *Блок Powered by Drupal (Под управлением Drupal) в новой области, находящейся над содержимым*

1. В первую очередь необходимо сообщить системе Drupal о новой области. Откройте файл *newmarine.info* снова и определите области. Не забывайте: чтобы добавить новую область, необходимо определить

все необходимые имеющиеся области, включая области по умолчанию. Добавьте в конец файла следующие строки:

```
regions[header] = Header
regions[left] = Left sidebar
regions[right] = Right sidebar
regions[content_top] = Content Top
regions[content] = Content
regions[footer] = Footer
```

2. Затем необходимо очистить кэш темы, чтобы система Drupal могла отобразить произведенные нами изменения. Перейдите на страницу Administer (Администрирование)→Site configuration (Настройка сайта)→Performance (Производительность) и щелкните на кнопке Clear cached data (Очистить кэш данных).
3. Перейдите на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Blocks (Блоки) (*admin/build/block*). Когда вы взглянете на список областей, среди них вы найдете новую область с названием «Content Top» (Над содержанием), куда мы можем поместить наши блоки. Переместите блок Powered by Drupal (Под управлением Drupal) из области Footer (Подвал) в область Content Top (Над содержанием) и щелкните на кнопке Save blocks (Сохранить блоки).
4. Вы увидите, что теперь блок исчез совсем. Мы убрали блок из области Footer (Подвал), но, несмотря на наличие области Content Top (Над содержанием), система Drupal пока не знает, где отобразить ее. Нам необходимо добавить информацию о новой области в файл шаблона, поэтому откройте файл *page.tpl.php* темы Newmarine.
5. Нам требуется добавить область в начало элемента div с идентификатором «main», выше существующей области content (содержание). Для этого вставьте строку, как показано ниже в строке 42:

```
41 <div id="main">
42   <?php print $content_top; ?>
43   <h1 class="title"><?php print $title ?></h1>
44   <div class="tabs"><?php print $tabs ?></div>
45   <?php if ($show_messages) { print $messages; } ?>
46   <?php print $help ?>
47   <?php print $content; ?>
48   <?php print $feed_icons; ?>
49 </div>
```

6. Сохраните файл и обновите страницу. Теперь вы должны увидеть логотип блока Powered by Drupal (Под управлением Drupal) над содержанием.

В центре внимания: дополнительные возможности переопределения

Чтобы получить возможность добраться до динамических элементов и управлять ими, необходимо выйти за пределы HTML и CSS и приступить к использованию PHP – только на языке PHP можно оказывать воздействие на глубинные слои Drupal. В следующих двух разделах рассказывается о том, как переопределять переменные и функции в файлах *template.php* тем оформления. Чтобы извлечь максимум пользы из этой части главы, вы должны иметь представление о переменных и функциях в языке PHP.

Намного больше информации о таких расширенных возможностях переопределения можно найти в руководстве по созданию тем в Drupal 6: *http://drupal.org/node/173880*.

Переменные шаблона

Те небольшие объемы динамического содержимого, которые используются в файлах шаблонов, обеспечивают немалые удобства – если не вызывает раздражения, что вывод получается не совсем такой, как хотелось бы. Переменные, создаваемые системой Drupal по умолчанию, такие как `$site_name` и `$submitted`, образуют основу, которая в большинстве случаев соответствует потребностям разработчика. Однако, как и для всего остального в механизме поддержки тем оформления, имеется возможность вскрыть доступ к ним и изменить их. Кроме этого, вы можете не только изменять существующие переменные, но и создавать свои собственные.

Все переменные, которые попадают в файл шаблона, сначала передаются специальной функции с именем `preprocess`. Вы можете добавить в файл *template.php* собственную функцию `preprocess` и воспользоваться возможностью внести последние изменения в переменные, прежде чем они попадут в шаблон. Для большей четкости можно использовать по одной функции `preprocess` на каждый шаблон, например, можно создать функцию `mytheme_preprocess_page`, чтобы воздействовать на переменные, используемые в файле шаблона *page.tpl.php*. С помощью этой функции можно определять или переопределять любые переменные. Например, если необходимо создать переменную для вывода случайных значений в диапазоне от 1 до 100, ее можно добавить в файле *template.php*:

```
function mytheme_preprocess_page(&$vars) {
  $vars['random_number'] = rand(1, 100);
}
```

Затем (после очистки кэша Drupal) значение этой переменной можно было бы вывести в файле *page.tpl.php*, как значение любой другой переменной:

```
<?php print $random_number; ?>
```

Переопределение существующей переменной выполняется точно так же. Когда вы присваиваете значение уже существующей в Drupal переменной, ваше значение затрет имеющееся. Подробнее об этом мы поговорим в практическом разделе, где покажем, как переопределить переменную `$submitted`.

Функции темы оформления

Значительная часть разметки HTML, которая выводится системой Drupal, находится в легкодоступном виде в файлах шаблонов. Но как быть с навигационными цепочками, которые отображают путь к текущей странице? К сожалению, нет никакого файла *breadcrumb.tpl.php*. Чтобы изменить что-то в разметке навигационной цепочки, необходимо погрузиться немного глубже, до уровня, где в Drupal определяется остальная часть разметки, который называется *функции темы оформления*.

Функции темы оформления – это обычные функции на языке PHP в программном коде Drupal, имена которых начинаются с `theme_`. Каждый элемент страницы, отсутствующий в файле шаблона, создается функцией темы оформления, такой как `theme_breadcrumb()`:

```
function theme_breadcrumb($breadcrumb) {
  if (!empty($breadcrumb)) {
    return '<div class="breadcrumb">'. implode(' » ', $breadcrumb) .'</div>';
  }
}
```

В таком виде данная функция будет выводить на странице навигационную цепочку, такую как Home » Administer » Site building » Themes. Но как быть, если потребуется, чтобы навигационная цепочка выводилась как Home::Administer::Site building::Themes, где элементы цепочки отделяются друг от друга двумя символами двоеточия? Нет проблем! Мы можем открыть файл *theme.inc* и внести необходимые изменения, так ведь? Нет, не так!

Функция theme()

Применительно к функциям тем оформления имеется одно важное замечание – они никогда не вызываются непосредственно. Они всегда вызываются с помощью центральной функции `theme()`. Поэтому, даже при том, что модуль может определить функцию, например `theme_username()`, когда ему потребуется вывести свой код разметки, он должен вызывать функцию `theme('username')`. Функция `theme()` является механизмом, обеспечивающим возможность переопределения.

Правильный способ переопределения функций тем оформления состоит в том, чтобы скопировать их в свой файл *template.php* и присвоить им имена в соответствии с соглашениями об именованиях, принятыми в Drupal, чтобы система могла их обнаруживать. После этого вы можете вносить любые необходимые изменения. На рис. 11.17 иллюстрируется, как Drupal определяет, какую функцию следует вызвать для отображения элемента страницы.

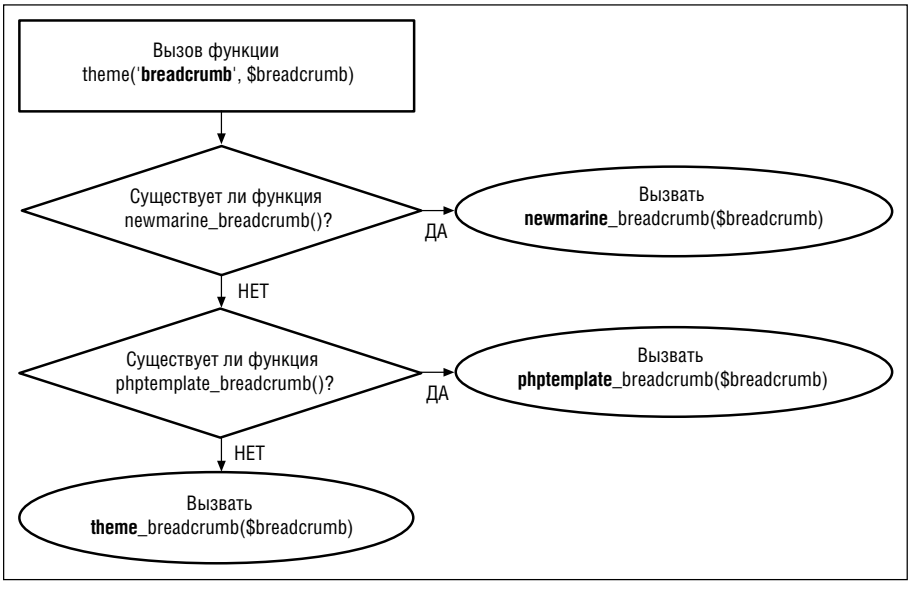


Рис. 11.17. *Как Drupal определяет, какую функцию вызвать*

Система Drupal выполняет обход целой иерархии имен, чтобы определить, какую версию функции в конечном итоге следует вызвать. Она выясняет, какой идентификатор использовать в качестве префикса в имени функции. Существует три возможных варианта, по возрастанию степени специализации: `theme`, *enginename* и *themename*, где *enginename* – это название используемого механизма шаблонов (наиболее часто в Drupal для работы с темами оформления используется механизм PHPTemplate), а *themename* – это буквальное название темы, как оно определено в файле *.info*. Система Drupal старается сначала определить наличие функции с самым специализированным именем (*themename*), а в случае неудачи продолжает поиск далее, вверх по списку. Она будет использовать первое найденное имя. Это означает, что если вы желаете использовать в своей теме собственную версию функции, то вам достаточно создать функцию с корректным именем; после этого система Drupal будет использовать вашу версию вместо оригинальной.

Так, в данном примере функция `theme_breadcrumb()` могла бы превратиться в функцию `newmarine_breadcrumb()`. Именно это мы и сделаем в следу-

ющем разделе «Практика: использование файла template.php для переопределения». Дополнительные сведения о функциях тем оформления можно отыскать в справочном руководстве Drupal, по адресу *http://drupal.org/node/11811*.

Практика: использование файла template.php для переопределения

Переопределение переменной шаблона

Drupal предоставляет в наше распоряжение множество переменных, но они не всегда отображаются как нам хотелось бы. Наиболее часто многие хотели бы изменить строку «Submitted by...» («Отправлено...»), в которой выводится имя автора и дата создания материала. Включить или выключить отображение этой строки можно на странице Administer (Администрирование)→Site building (Конструкция сайта)→Themes (Темы оформления) (*admin/build/themes*), на вкладке Configure (Настроить) (*admin/build/themes/settings*), но нет никакой возможности изменить фактическую информацию, которая выводится в этой строке. Мы изменим эту строку так, чтобы в ней выводился текст «Posted on» («Опубликовано»), за которым следует дата публикации, как показано на рис. 11.18.

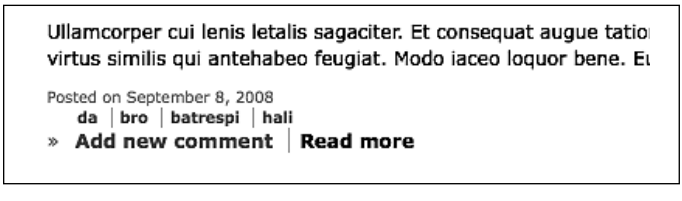


Рис. 11.18. *Текст «Submitted by...» (Отправлено...) замещен текстом «Posted on...» («Опубликовано...»)*

1. Чтобы изменить этот текст, нам следует модифицировать файл *template.php*. А так как в нашей теме Newmarine этот файл отсутствует, чтобы решить поставленную задачу, мы создадим его. Если перейти к узлу, отображающему опубликованную информацию, и произвести те же манипуляции с помощью блока Drupal Themer Information (Информация для разработчика тем), которые мы выполняли, когда двигались к переменной `$breadcrumb`, можно обнаружить родительский файл шаблона для строки «Submitted by...» («Отправлено...»). Вы без труда узнаете, что родительским является файл *node.tpl.php*. Если открыть этот файл (или *nodestory.tpl.php*, если вы следовали за практическим примером в предыдущем разделе «Оформление отдельных типов содержимого»), можно увидеть, что в нем присутствует инструкция вывода переменной `$submitted`, которая воспроизводит текст, как показано ниже (вспомните, как ранее мы переместили эту строку вместе с выводом терминов таксономии):

```
<?php if ($page == 0) { ?><h2 class="title"><a href=
"<?php print $node_url?>"><?php print $title?></a></h2><?php }; ?>
<div class="content"><?php print $content?></div>
<span class="submitted"><?php print $submitted?></span>
<div class="taxonomy"><?php print $terms?></div>
```

2. Необходимые изменения мы можем внести в нашем файле *template.php*. Но тема Newmarine еще не имеет файла *template.php*. Поэтому создадим его и поместим в папку *newmarine*, рядом с другими файлами темы.
3. После создания файла *template.php* необходимо добавить в него новую функцию preprocess, чтобы получить доступ к переменной, которую требуется изменить. Добавьте следующие строки в файл *template.php*:

```
<?php
// $Id:$
/**
 * Реализация функции hook_preprocess_node().
 */
function newmarine_preprocess_node(&$vars) {

}
```

4. Теперь необходимо фактически изменить переменную, для чего внутри функции добавим переменную и запишем в нее требуемое значение:

```
<?php
// $Id:$
/**
 * Реализация функции hook_preprocess_node().
 */
function newmarine_preprocess_node(&$vars) {
  // Изменить значение переменной $submitted так, чтобы выводился текст,
  // такой как "Posted on June 12, 2008".
  $vars['submitted'] = t('Posted on ') .
  format_date($vars['node']->created, 'custom', 'F j, Y');
}
```

5. Опять же не забудьте очистить кэш на странице Administer (Администрирование)→Site configuration (Настройка сайта)→Performance (Производительность) (*admin/settings/performance*).
6. После обновления страницы вы должны увидеть строку с информацией о дате публикации в новом формате.

Переопределение функции темы оформления

Переопределение функций темы оформления обычно выполняется простым копированием соответствующих функций в файл *template.php*

темы. Основная сложность состоит в том, чтобы *отыскать* требуемую функцию темы. Мы собираемся изменить внешний вид навигационной цепочки так, чтобы ее элементы отделялись друг от друга не символом », а парой символов :: (двумя двоеточиями), и получить результат, изображенный на рис. 11.19.

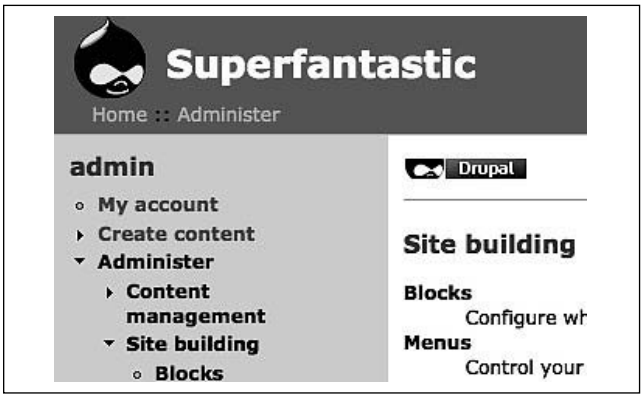


Рис. 11.19. *Навигационная цепочка, которая воспроизводится функцией newmarine_breadcrumb()*

Мы снова воспользуемся информацией, которую поставляет инструмент Devel Themer Information, для чего перейдем на страницу, где присутствует навигационная цепочка. Вполне подойдет такая страница как Administer (Администрирование)→Site building (Конструкция сайта) (*admin/build*). Выполните следующие действия:

1. Включите флажок Themer info (Информация о теме). Наведите указатель мыши на навигационную цепочку и щелкните на ней, чтобы отобразить информацию в окне Drupal Themer Information (Информация для разработчика тем).
2. На этот раз вместо раздела Parents info (Информация о родителе) нам необходимо найти раздел Function called: (Вызываемая функция). В нем крупным шрифтом должно быть указано имя `theme_breadcrumb()`, как показано на рис. 11.20.
3. Теперь нам известно, какая функция воспроизводит навигационные цепочки, и мы можем скопировать ее. Откройте файл *includes/theme.inc* и отыщите функцию `theme_breadcrumb`. Скопируйте функцию `theme_breadcrumb` целиком. Она просто принимает массив ссылок HTML и с помощью функции `implode()` объединяет их в одну строку, используя символ », обертывая ее тегом `<div>` с классом `breadcrumb`.
4. Откройте свой файл *template.php* и вставьте в него скопированную функцию. Переименуйте ее в `newmarine_breadcrumb`, как показано ниже:

```
/**
 * Возвращает навигационную цепочку с требуемым оформлением.
 *
 * @param $breadcrumb
 *   Массив, содержащий ссылки для навигационной цепочки.
 * @return строку, содержащую навигационную цепочку.
 */
function newmarine_breadcrumb($breadcrumb) {
  if (!empty($breadcrumb)) {
    return '<div class="breadcrumb">'. implode(' » ', $breadcrumb) .'</div>';
  }
}
```

Если вы забудете переименовать функцию, то при попытке использовать ее вы получите сообщение об ошибке: «Fatal error: Cannot redeclare theme_breadcrumb()» (Фатальная ошибка: повторное объявление функции theme_breadcrumb()). Все не так страшно! Дело в том, что ядро Drupal уже использует имя функции `theme_breadcrumb`, поэтому вы не можете вторично использовать его. Просто вернитесь к файлу *template.php* и измените имя функции в соответствии с названием темы. Обновите страницу, и ваш сайт опять заработает.

Рис. 11.20. *Модуль Theme Developer показывает, какая функция используется для вывода навигационных цепочек*

5. Теперь нужно просто заменить символ » парой символов двоеточий (::), как показано ниже:

```
function newmarine_breadcrumb($breadcrumb) {
  if (!empty($breadcrumb)) {
    return '<div class="breadcrumb">'. implode(' :: ', $breadcrumb) .'</div>';
  }
}
```

6. Очистите кэш и обновите страницу. Теперь навигационная цепочка выглядит так, как нам и хотелось.

Дополнительно

Знание языка PHP поможет вам добиться многого. В справочном руководстве имеется целый раздел (*http://drupal.org/handbook/customization/snippets*), где приводятся интересные фрагменты программного кода, позволяющие решать различные задачи. Эти фрагменты обеспечивают самый простой способ привести сайт в соответствие со своими потребностями.

Многие фрагменты связаны с копированием и вставкой программного кода PHP в такие элементы, как узлы и блоки. Чтобы обезопасить себя, необходимо задействовать модуль PHP Filter, входящий в состав ядра Drupal. Будьте *крайне осторожны* при передаче прав доступа к этому фильтру – в идеале этим правом должен обладать только пользователь №1, и никто другой. За дополнительной информацией об обеспечении безопасности с помощью механизма фильтрации обращайтесь к главе 2.

Фрагменты программного кода, которые приводятся в справочном руководстве, получены от пользователей Drupal, то есть любой, имеющий учетную запись на сайте Drupal.org, сможет добавлять свои фрагменты. Они не просматриваются и не модерируются, поэтому вы можете использовать их только на свой страх и риск! Вы всегда должны внимательно просматривать фрагменты, чтобы убедиться в отсутствии скрытых ловушек. Обязательно сначала проверяйте их и составляйте свое мнение о безопасности в соответствии с информацией на странице *http://drupal.org/writing-secure-code*.

Все фрагменты делятся на несколько категорий. Наиболее часто используются фрагменты из категорий Theme snippets (Фрагменты тем) (*http://drupal.org/node/45471*) и PHP snippets (Фрагменты программного кода PHP) (*http://drupal.org/handbook/customization/php-snippets*). В разделе Theme snippets (Фрагменты тем) приводятся фрагменты программного кода, использующие методы, описанные в этой главе, и помогающие настроить внешний вид сайта. Раздел PHP snippets (Фрагменты программного кода PHP) в свою очередь делится на подразделы, наиболее важные из которых перечислены ниже:

Mini modules (Мини-модули)

Здесь перечислены очень простые модули, которые могут копироваться и вставляться при решении определенных задач. Инструкции по использованию этих мини-модулей можно найти по адресу *http://drupal.org/node/70903*.

PHP block snippets (Фрагменты блоков на языке PHP)

Эти фрагменты реализуют специализированные блоки, использующие модуль PHP filter, – они позволяют создавать свои собствен-

ные динамические блоки и помещать их в любое место на страницах сайта.

PHP block visibility settings (Настройки видимости блоков на языке PHP)

В противоположность фрагментам «содержимого» блоков, перечисленным выше, эти фрагменты управляют видимостью блоков для посетителей. В обычном случае у вас имеется возможность настраивать видимость блоков, определяя с помощью флажков роли и страницы. Эти фрагменты обеспечивают более широкий диапазон критериев управления видимостью.

PHP page snippets (Фрагменты страниц)

Эти фрагменты применяются для создания типов содержимого, использующих модуль PHP filter, и позволяют генерировать динамические узлы. Вообще говоря, применение этих фрагментов не приветствуется – если вам действительно необходимо создать собственные динамические страницы, то лучше сделать еще один шаг и создать модуль, напоминающий мини-модуль. Они просты в создании и не требуют использования модуля PHP filter, работа с которым должна быть максимально ограничена из-за проблем с безопасностью сайта.

В заключение

Мы охватили лишь вершину айсберга возможностей по созданию и настройке тем оформления сайта, тем не менее мы продемонстрировали все основные понятия. Мы показали, как изменять стили CSS в теме, как можно изменить структуру страницы и узла содержимого, а также – как переопределять файлы шаблонов, переменные и функции тем оформления. Хотелось бы надеяться, что приведенные примеры помогут вам понять, какая мощь скрыта в механизме поддержки тем оформления системы Drupal. Существует еще целая масса различных способов изменения разметки в темах оформления, генерируемых системой Drupal, а также – создания правил, определяющих порядок структурирования страниц и информационного наполнения, в зависимости от области, в которой они отображаются, да и вообще от любых других критериев.

Желающие овладеть мастерством создания тем оформления найдут дополнительную информацию, касающуюся тем, на странице *http://drupal.org/theme-guide* и в различных печатных изданиях, посвященных созданию тем в Drupal.

Механизм поддержки тем оформления в Drupal стремится обеспечить максимальную гибкость, позволяя опытным разработчикам изменять любые аспекты страниц самыми разными способами. И все-таки – это всего лишь набор инструментов. Как и весь веб-дизайн, разработка тем

оформления в Drupal ограничивается лишь творческими способностями и навыками тех, кто занимается их реализацией. В умелых руках система Drupal позволяет создавать настолько привлекательные и функциональные веб-сайты, насколько только это возможно представить.

Ссылки

Ниже перечислены темы и модули, упоминавшиеся в этой главе:

- Тема Bluemarine: ядро
- Модуль Devel: *http://drupal.org/project/devel*
- Тема Foundation: *http://drupal.org/project/foundation*
- Тема Skyliner: *http://drupal.org/project/skyliner*
- Тема Zen: *http://drupal.org/project/zen*

Ниже перечислены ресурсы на сайте Drupal.org, упоминавшиеся в этой главе:

- Шаблоны и рекомендации по умолчанию: *http://drupal.org/node/190815*
- Ресурсы для разработчиков Drupal: *http://drupal.org/node/147789*
- Руководство по разработке тем Drupal: *http://drupal.org/theme-guide*
- Сторонние темы для Drupal: *http://drupal.org/project/Themes*
- Сторонние механизмы поддержки тем в Drupal: *http://drupal.org/project/Theme+Engines*
- Переопределение функций тем оформления: *http://drupal.org/node/11811*
- Стандарт использования тега PHP: *http://drupal.org/node/545*
- Переменные механизма PHPTemplate: *http://drupal.org/phptemplate*
- Руководство по обеспечению безопасности программного кода: *http://drupal.org/writing-secure-code*
- Фрагменты: *http://drupal.org/handbook/customization/snippets*
- Фрагменты на языке PHP: *http://drupal.org/handbook/customization/php-snippets*
- Фрагменты тем оформления: *http://drupal.org/node/45471*
- Руководство по обновлению тем: *http://drupal.org/update/theme*

Ниже перечислены внешние ресурсы, упоминавшиеся в этой главе:

- Расширение Firebug: *https://addons.mozilla.org/en-US/firefox/addon/1843*
- Текстовый редактор gedit: *http://www.gnome.org/projects/gedit*

- Библиотека jQuery: *http://jquery.com*
- Текстовый редактор Kate: *http://kate-editor.org*
- Текстовый редактор Notepad++: *http://notepad-plus.sourceforge.net*
- Текстовый редактор Smultron: *http://smultron.sourceforge.net*
- Текстовый редактор TextWrangler: *http://www.barebones.com/products/textwrangler*
- Панель инструментов веб-разработчика: *https://addons.mozilla.org/en-US/firefox/addon/60*
- W3Schools: *http://w3schools.com*

A

Установка и обновление Drupal

Первый шаг на пути к использованию Drupal, естественно, заключается в том, чтобы получить необходимое программное обеспечение и установить его. Система Drupal распространяется в комплекте со сценарием установки, который проведет вас через несколько страниц сбора необходимой информации, затем создаст базу данных и файл с настройками сайта. Мы рассмотрим все операции, которые вам необходимо будет выполнить, чтобы установка прошла без сучка и задоринки, – вы увидите, что установка Drupal выполняется быстро и просто, как только будет собрана вся необходимая информация.

После установки и запуска системы Drupal очень важно обеспечить своевременное обновление программного обеспечения сайта. Периодически выходят новые версии сторонних модулей и ядра Drupal, в которых устраняются критические проблемы, связанные с безопасностью, поэтому очень важно своевременно выполнять обновления по мере их выхода. Мы рассмотрим модуль Update Status, встроенный в Drupal 6, который будет извещать вас о появлении обновлений для вашего сайта, а также поговорим о том, как выполнять обновление отдельных модулей и самого ядра Drupal одной версии к другой.

Вы можете заметить, что во многих случаях (и даже в документации, описывающей ядро Drupal) термины «обновление» (updating) и «модернизация» (upgrading) используются, как взаимозаменяемые. Оба эти термина обозначают замену существующего кода более новым кодом.

Перед установкой

Перед установкой Drupal важно убедиться, что вы действительно можете ее выполнить, для чего требуется иметь некоторое представление о структуре системы Drupal. В этом разделе мы рассмотрим перечень

требований, предъявляемых системой Drupal, и остановимся на некоторых важных аспектах структуры дерева каталогов Drupal, которые желательно знать перед началом процесса установки.

Сбор необходимой информации

Перед установкой Drupal очень важно убедиться в соответствии определенным требованиям. Полный перечень требований приводится на странице http://drupal.org/requirements. Ниже приводится перечень основных требований, соответствие которым следует проверить перед установкой Drupal:

1. Наличие доступа к веб-хосту или к локальной среде разработки, со следующими возможностями:
   a. *Веб-сервер*, такой как Apache (*http://httpd.apache.org*), способный передавать страницы Drupal броузеру. Кроме того, наличие доступа к расширению *mod_rewrite* веб-сервера Apache позволит задействовать особенность Drupal «Clean URLs» (Чистые ссылки), которая выполняет приведение адресов URL вида *http://www.example.com/index.php?q=contact* к виду *http://www.example.com/contact*.
   b. *PHP* (*http://php.net*), динамический язык сценариев, являющийся движущей силой Drupal. Drupal 6 требует наличие поддержки PHP версии не ниже 4.3.5, при этом рекомендуется использовать версию PHP 5.2 или выше. На странице требований, на сайте Drupal.org, дается более полная информация об обязательных и рекомендуемых расширениях PHP, большая часть из которых включена по умолчанию.
   c. *Сервер баз данных*, такой как MySQL (*http://mysql.com*), где система Drupal будет хранить все содержимое сайта, свои данные и настройки, необходимые для нормальной работы.

В данной книге предполагается, что вы используете Apache и MySQL. За дополнительной информацией о поддержке других веб-серверов и баз данных обращайтесь по адресу *http://drupal.org/getting-started/6/install*.

2. Выпишите на листок бумаги следующую информацию о своем веб-хосте:
   a. Ваши имя пользователя и пароль для (S)FTP или SSH, с помощью которых вы сможете записать файлы Drupal к себе.
   b. Сведения о своем сервере баз данных, включая имя пользователя, пароль и название базы данных, – чтобы обеспечить Drupal возможностью соединения с базой данных. Кроме того, для некоторых веб-хостов необходимо указать дополнительные параметры

соединения с базой данных, например указать имя удаленного хоста или порт подключения к серверу баз данных.

3. Прежде чем начинать установку Drupal, вам также потребуется создать базу данных, куда будет выполняться установка, – сама система Drupal не создает базу данных, так как для этой операции обычно требуются «повышенные» привилегии доступа к серверу баз данных. Систему Drupal можно установить либо в отдельную базу данных, либо в общую базу данных, которая используется также другими приложениями, указав Drupal-префикс для имен таблиц; но вообще лучше, если для Drupal будет создана отдельная база данных. Обратитесь к своему поставщику услуг хостинга или к системному администратору, если вам необходима дополнительная информация о порядке создания новой базы данных, и выпишите ее название, так как оно потребуется немного позже. Кроме того, не забудьте выписать имя пользователя и пароль для доступа к базе данных.

Для нужд разработки будет гораздо проще развернуть локальную веб-среду, чтобы иметь возможность вносить изменения до их выгрузки в окончательное местоположение. Существует несколько свободных пакетов программного обеспечения, реализующих более или менее нетребовательные решения, включая XAMPP (*http://www.apachefriends.org/en/xampp.html*) для Windows и Linux, WampServer (*http://www.wampserver.com*) для Windows и MAMP (*http://mamp.info/en/download.html*) для Macintosh.

Как только вы убедитесь, что все необходимое у вас уже имеется, можно приступать к установке.

Загрузка Drupal

Первый шаг перед установкой заключается в том, чтобы получить сам программный код Drupal. Вы можете получить исходный программный код Drupal на сайте книги *http://usingdrupal.com/source_code* или загрузить его непосредственно на сайте Drupal.org. Ниже описываются шаги, которые следует выполнить для загрузки программного кода с сайта Drupal.org:

1. Перейдите на страницу *http://drupal.org* и найдите несколько ссылок для загрузки Drupal. Они выделены на рис. А.1. Щелкните на вкладке Download (Загрузить) в правом верхнем углу страницы.
2. На следующей странице перечислены все типы проектов, доступных для загрузки: модули, темы оформления, переводы и так далее. Щелкните на ссылке Drupal project (Проект Drupal), чтобы перейти на страницу ядра Drupal.
3. В таблице, изображенной на рис. А.2, перечислены доступные версии в порядке от более новых к более старым. Если только вы не оказываете помощь в разработке Drupal, вам следует загружать версии,

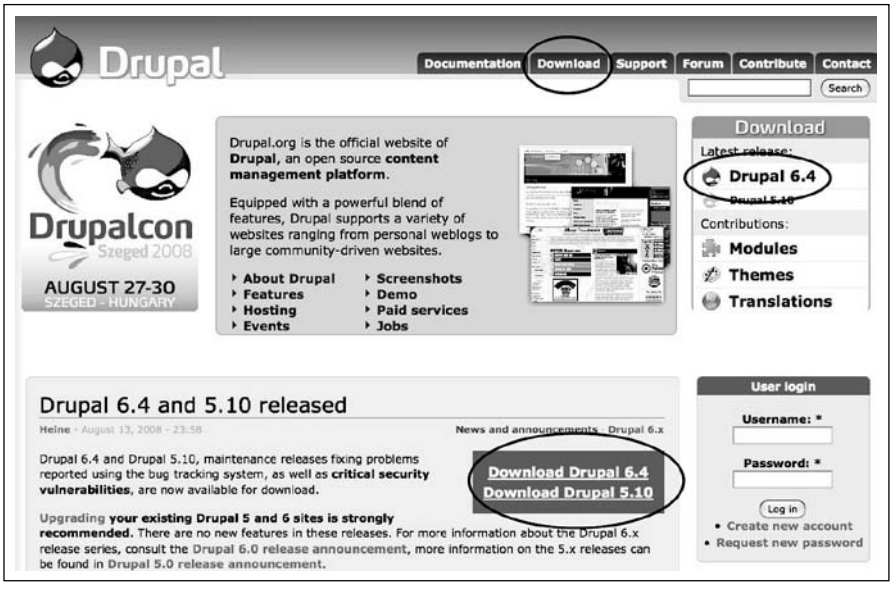
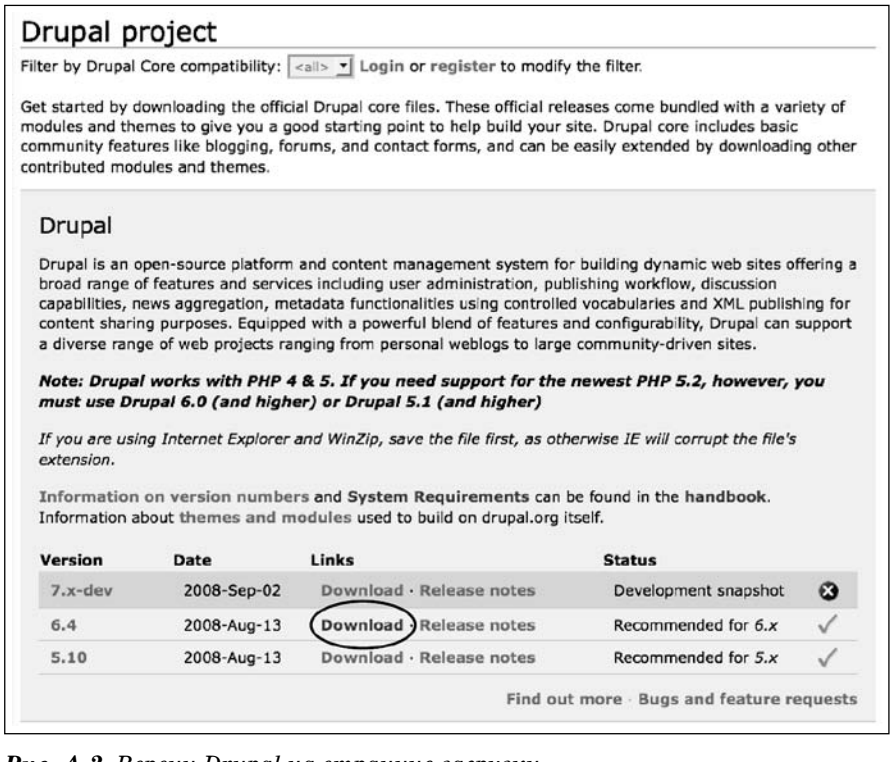


Рис. А.1. *Ссылки для загрузки на сайте Drupal.org*



Drupal project

Filter by Drupal Core compatibility: <all> Login or register to modify the filter.

Get started by downloading the official Drupal core files. These official releases come bundled with a variety of modules and themes to give you a good starting point to help build your site. Drupal core includes basic community features like blogging, forums, and contact forms, and can be easily extended by downloading other contributed modules and themes.

Drupal

Drupal is an open-source platform and content management system for building dynamic web sites offering a broad range of features and services including user administration, publishing workflow, discussion capabilities, news aggregation, metadata functionalities using controlled vocabularies and XML publishing for content sharing purposes. Equipped with a powerful blend of features and configurability, Drupal can support a diverse range of web projects ranging from personal weblogs to large community-driven sites.

Note: Drupal works with PHP 4 & 5. If you need support for the newest PHP 5.2, however, you must use Drupal 6.0 (and higher) or Drupal 5.1 (and higher)

If you are using Internet Explorer and WinZip, save the file first, as otherwise IE will corrupt the file's extension.

Information on version numbers and **System Requirements** can be found in the **handbook**. Information about themes and modules used to build on drupal.org itself.

Version	Date	Links	Status
7.x-dev	2008-Sep-02	Download · Release notes	Development snapshot
6.4	2008-Aug-13	Download · Release notes	Recommended for *6.x*
5.10	2008-Aug-13	Download · Release notes	Recommended for *5.x*

Find out more · Bugs and feature requests

Рис. А.2. *Версии Drupal на странице загрузки*

помеченные как Recommended (Рекомендуется). Это так называемые «стабильные версии». Чтобы иметь возможность использовать примеры из этой книги, щелкните на ссылке Download (Загрузить) для версии, помеченной как Recommended for 6.x (Рекомендуется для версии 6.x).

4. Файлы Drupal упакованы с помощью утилиты tar и сжаты с помощью утилиты gzip. В результате этого файл пакета имеет расширение *tar.gz*. Такие файлы очень напоминают архивы, созданные программой zip. Сохраните файл пакета и затем извлеките его содержимое с помощью предпочитаемого вами приложения для работы с архивами.
5. Поместите распакованные файлы на веб-сервер с помощью программы-клиента (S)FTP или зарегистрировавшись на сервере, и загрузите пакет непосредственно на сервер и распакуйте его там.

Файлы и каталоги Drupal

Теперь, когда вы загрузили пакет Drupal, следует потратить несколько минут, чтобы осмотреть его содержимое. Знакомство с основной структурой размещения важных файлов и каталогов поможет приоткрыть завесу тайны и получить некоторое представление о том, как все это действует. Когда вы откроете папку Drupal, вы увидите дерево каталогов, как показано на рис. А.3.

Самыми важными для рассмотрения в этом разделе являются файлы установки и обновления, а также каталог *sites*. Файлы *install.php* и *update.php* – это два сценария, которые фактически выполняют операции, соответствующие их названиям. Так как они находятся в папке самого верхнего уровня, которая также называется корневым каталогом Drupal, вы можете обратиться к ним напрямую, введя в адресной строке броузера примерно такой адрес: *http://example.com/install.php*. Помимо сценариев здесь также присутствуют два текстовых файла, по одному для каждой операции: *INSTALL.txt* и *UPGRADE.txt*. Эти файлы, которые мы также будем рассматривать в этом приложении, содержат инструкции по использованию соответствующих им сценариев.

Большинство начинающих администраторов Drupal, взглянув на структуру каталогов, изображенную на рис. А.3, решат размещать сторонние и собственные модули и темы оформления в каталоги *modules* и *themes*, находящиеся в корневом каталоге Drupal, соответственно. Именно в них хранятся базовые модули и темы Drupal, поэтому их и нужно использовать, разве не так? Если вы поместите свои файлы в эти каталоги, они будут работать и Drupal будет в состоянии обнаружить их. Однако это приведет к появлению проблем при первой же попытке обновления до следующей версии с исправлениями в системе безопасности, так как перезаписывание этих каталогов новыми базовыми версиями приведет к уничтожению всех ваших изменений и модификаций. Лучше всего сохранять все сторонние и собственные модули

и темы в каталоге *sites*. Это означает, что вы должны создать новые каталоги *modules* и *themes* в каталоге *sites/all*, если только вы не создаете сложную архитектуру с несколькими сайтами (подробнее об этом рассказывается во врезке ниже), и помещать сторонние и собственные разработки в эти каталоги, как показано на рис. А.4. При таком подходе все файлы, имеющие значение только для вашего сайта, будут храниться отдельно и не будут смешиваться с файлами ядра. Это *существенно* упростит процедуру последующих обновлений.

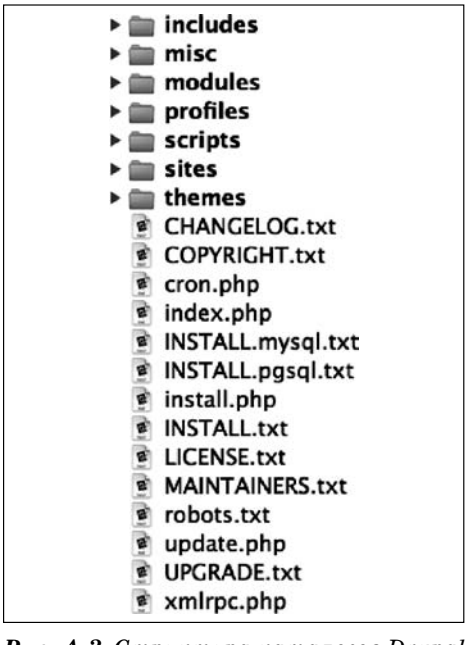


Рис. А.3. *Структура каталогов Drupal*

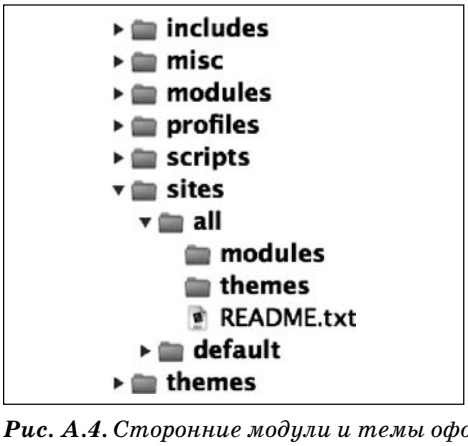


Рис. А.4. *Сторонние модули и темы оформления должны помещаться в каталог sites/all*

Несколько сайтов на одной установке Drupal

Одной из наиболее мощных особенностей Drupal является возможность запустить несколько веб-сайтов, совместно использующих один и тот же комплект файлов ядра системы. Эта возможность часто называется мультисайтингом. Эта возможность использована для реализации примеров веб-сайтов по адресу *http://www.usingdrupal.com* и позволяет избежать необходимости хранить несколько копий одних и тех же файлов.

Как действует эта возможность? На стороне Apache, в файле *httpd.conf*, для каждого поддомена создается запись о виртуальном хосте, указывающая на один и тот же комплект файлов Drupal, например:

```
<VirtualHost *>
  ServerName usingdrupal.com
  ServerAlias *.usingdrupal.com www.usingdrupal.com
  DocumentRoot /home/www/public_html
</VirtualHost>
```

Затем на стороне Drupal, внутри каталога *sites*, создаются каталоги для каждого дочернего сайта, в каждом из которых имеется свой собственный файл *settings.php* и каталог *files*. В результате такой организации настройки могли бы находиться в файле *sites/jumpstart.usingdrupal.com/settings.php*. Когда в броузере вводится адрес URL, такой как *http://jumpstart.usingdrupal.com*, Drupal отыщет в каталоге *sites* наиболее подходящий подкаталог и загрузит файл с настройками, находящийся в нем.

Однако поддержка нескольких сайтов не ограничивается лишь возможностью организовать несколько дочерних сайтов, как в данном случае. Совершенно разные веб-сайты также могут совместно использовать одну и ту же установку Drupal и при этом обладать собственными комплектами модулей и тем оформления. Можно даже создавать такие сложные конфигурации, когда одни и те же таблицы базы данных совместно используются различными сайтами, с целью обеспечить единство учетных записей или реализовать поиск по содержимому всех веб-сайтов.

За дополнительной информацией о возможности организации многосайтовых структур в Drupal обращайтесь к документации на сайте Drupal.org, по адресу *http://drupal.org/node/43816*.

Установка Drupal

Как только будут удовлетворены все требования и собрана вся необходимая информация, можно приступать к установке. Инструкции ниже

предполагают, что вы уже создали базу данных, загрузили Drupal и поместили файлы, извлеченные из архива, на веб-сервер:

1. Для корректной установки Drupal необходимо создать файл с настройками. Для этого внутри каталога Drupal создайте копию файла *sites/default/default.settings.php* с именем *sites/default/settings.php*, то есть уберите префикс *default.* в начале имени файла. Вам также необходимо будет обеспечить доступность нового файла *settings.php* для записи, например, командой `chmod 666` или `chmod a+w sites/default/settings.php`. На вашем веб-сервере должна иметься дополнительная информация о том, как обеспечить доступность файлов для записи. Загляните также в справочное руководство по системе Drupal, по адресу *http://drupal.org/node/202483*.

Файл с настройками должен именно *копироваться*, а не переименовываться. Системе Drupal требуются оба файла, чтобы создать корректный файл с настройками.

2. Теперь можно перейти по адресу *http://www.example.com/install.php*, чтобы начать процесс установки.
3. На первой странице мастера установки вы сможете выбрать язык, как показано на рис. А.5. По умолчанию доступен только английский язык. Однако вы можете загрузить другие переводы и выполнить установку Drupal на своем языке. В главе 8 вы найдете дополнительную информацию по установке и настройке многоязычных сайтов. Щелкните на ссылке Install Drupal in English (Установить Drupal на английском языке).

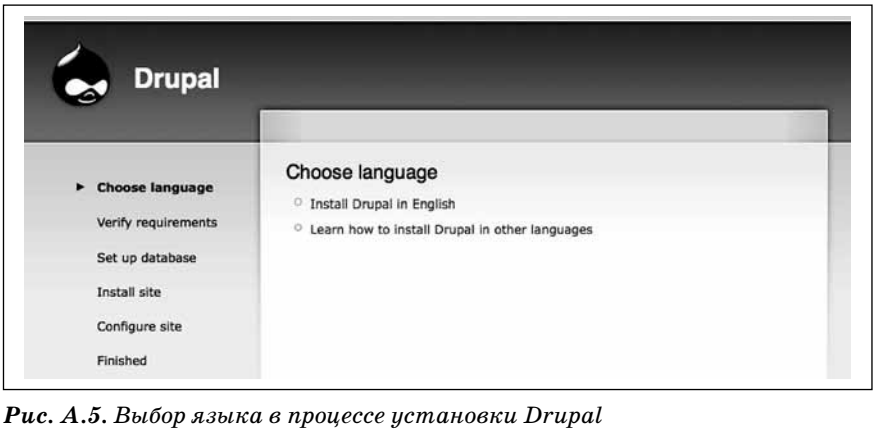


Рис. А.5. *Выбор языка в процессе установки Drupal*

Следующая страница сначала проверит наличие всех необходимых разрешений, прежде чем позволит продолжить процедуру установки. В зависимости от конфигурации хоста вам может потребоваться изменить права доступа к каталогу *sites*. Дополни-

тельную информацию вы можете получить на справочных страницах, ссылки на которые присутствуют в сообщениях об ошибках мастера установки.

4. Если все идет, как надо, вы должны увидеть страницу, где вам будет предложено указать информацию о базе данных, как показано на рис. А.6.

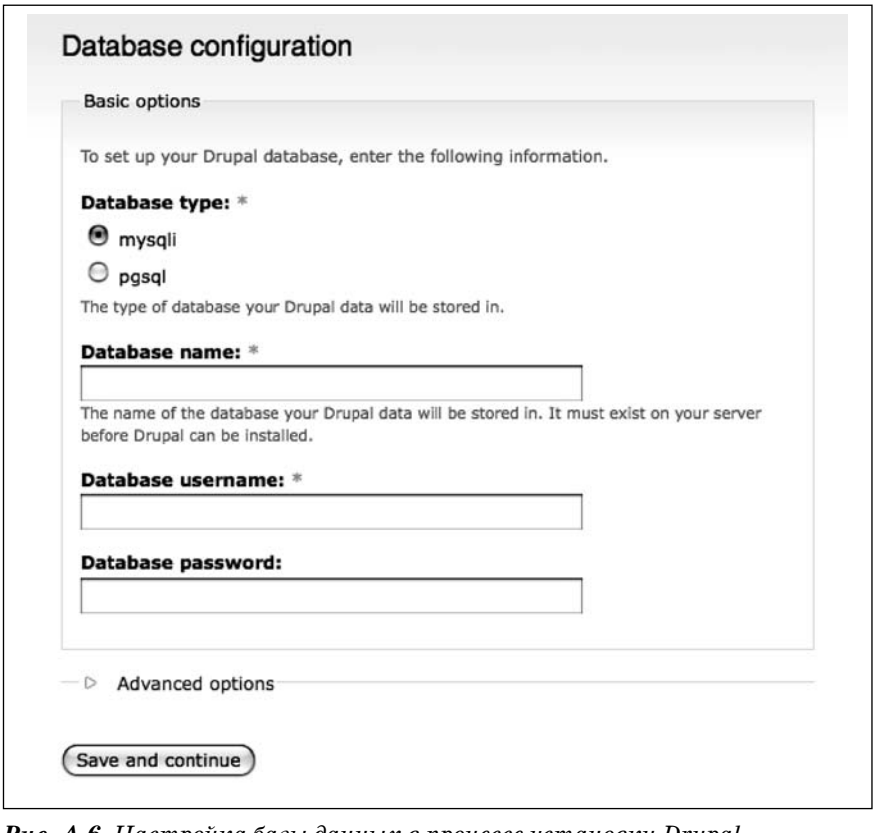


Рис. А.6. *Настройка базы данных в процессе установки Drupal*

5. Выше мы рекомендовали выписать на листе бумаги информацию о подключении к базе данных, включая имя пользователя и пароль. Теперь настало время использовать ее. Вам, как минимум, потребуется указать имя базы данных, имя пользователя базы данных и пароль. Если для вашего веб-сайта потребуется указать дополнительные сведения о базе данных, такие как имя хоста или порт подключения к серверу баз данных, разверните группу параметров Advanced options (Дополнительные настройки), чтобы ввести эти сведения. Как только вы введете всю необходимую информацию о базе данных, щелкните на кнопке Save and continue (Сохранить и продолжить).

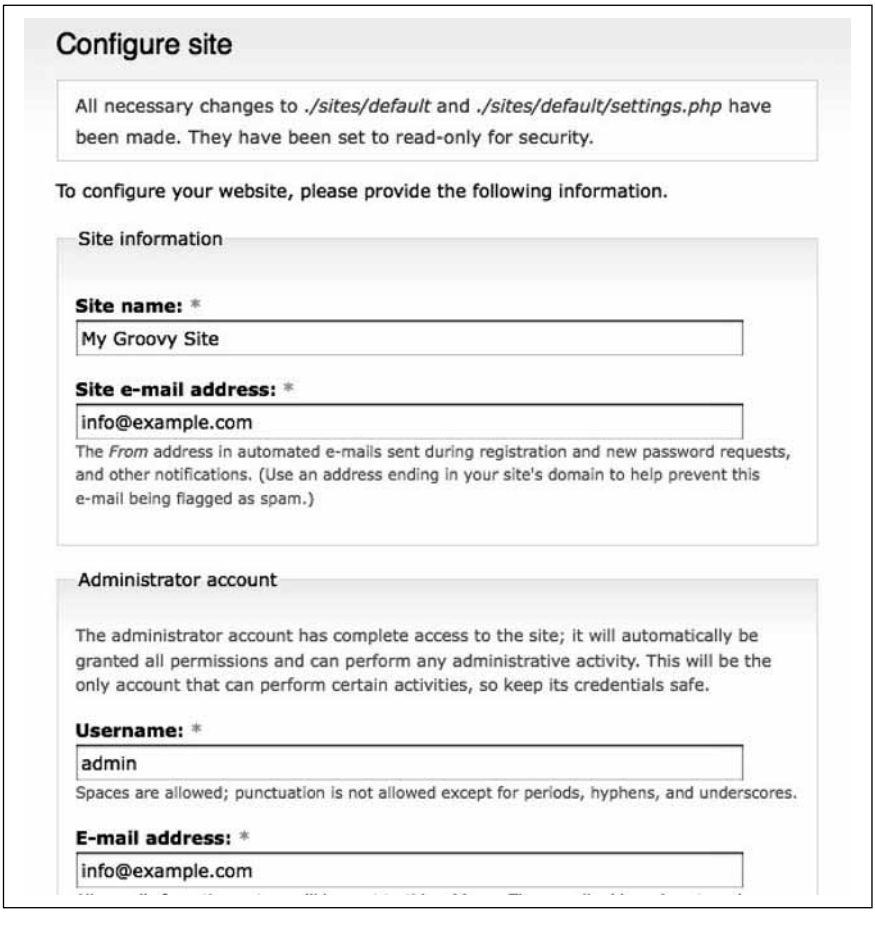


Рис. А.7. *Настройка параметров в процессе установки Drupal*

6. Следующая страница, как показано на рис. А.7, содержит список основных параметров сайта, которые требуется настроить.
7. Сначала следует заполнить группу параметров Site information (О сайте). В этой группе присутствуют следующие важные глобальные параметры сайта:

   Site name (Название сайта)

   Это название по умолчанию будет отображаться в полосе заголовка на всех страницах, а также в верхнем левом углу всех страниц.

   Site e-mail address (Адрес электронной почты сайта)

   Все системы электронной почты будут отправлять сообщения с этого адреса, например, при регистрации нового пользователя.
8. Следующий шаг – создание учетной записи администратора. Администратор (также известен, как «пользователь № 1») – это «супер-

пользователь», который освобождается от любых проверок на право доступа и имеет неограниченную власть над сайтом. Поэтому для данной учетной записи вы должны использовать пароль, который будет сложно подобрать (к счастью, Drupal попытается помочь вам, проверяя стойкость пароля по мере его ввода). Используйте эту учетную запись ответственно и только для решения административных задач. Для повседневного использования создайте вторую учетную запись с меньшим количеством привилегий.

9. Параметры в разделе Server settings (Настройки сервера) обычно можно оставить со значениями по умолчанию. В число этих параметров входят:

   Default time zone (Часовой пояс по умолчанию)

   Если пользователь не определит собственный часовой пояс в настройках своей учетной записи, все сообщения на сайте будут отображаться с указанием времени публикации в соответствии с указанным здесь часовым поясом. По умолчанию система Drupal установит часовой пояс, взяв его из настроек броузера, предполагая, что это совпадает с вашим желанием.

   Clean URLs (Чистые ссылки)

   Функция Clean URLs позволит вам иметь такие адреса URL, как *http://example.com/about*, вместо *http://example.com/?q=about*. Здесь будет выполнена проверка корректности настройки функции Clean URLs в веб-сервере, и если все будет в порядке, вы получите возможность включить этот параметр.

При возникновении проблем с поддержкой функции Clean URLs полезно иметь представление о том, что обычно эта проблема обусловлена следующими причинами:

- На веб-сервере не установлен модуль *mod_rewrite* (или эквивалентный ему).
- Веб-сервер не имеет возможности прочитать содержимое файла *.htaccess*, входящего в состав Drupal.

Попросите своего системного администратора выяснить, какая из причин сработала в вашем случае. За дополнительной информацией обращайтесь к разделу «Clean URLs» в электронном справочном руководстве, по адресу *http://drupal.org/node/15365*.

   Update notifications (Сообщения об обновлениях)

   Версия Drupal 6 поддерживает новую возможность – автоматически проверять наличие обновлений ядра системы, модулей и тем оформления и информировать вас об этом. *Настоятельно* рекомендуем включить этот параметр (включен по умолчанию), так как это позволит вовремя устанавливать обновления безопасности.

10. Как только будут введены все значения параметров, щелкните на кнопке Save and continue (Сохранить и продолжить).

11. Последняя страница сообщит вам, что установка завершена и можно приступать к настройке нового веб-сайта. Щелкните на ссылке your new site (ваш новый сайт), чтобы пуститься в путешествие по стране Drupal! На рис. А.8 показана начальная страница сайта Drupal сразу после установки.

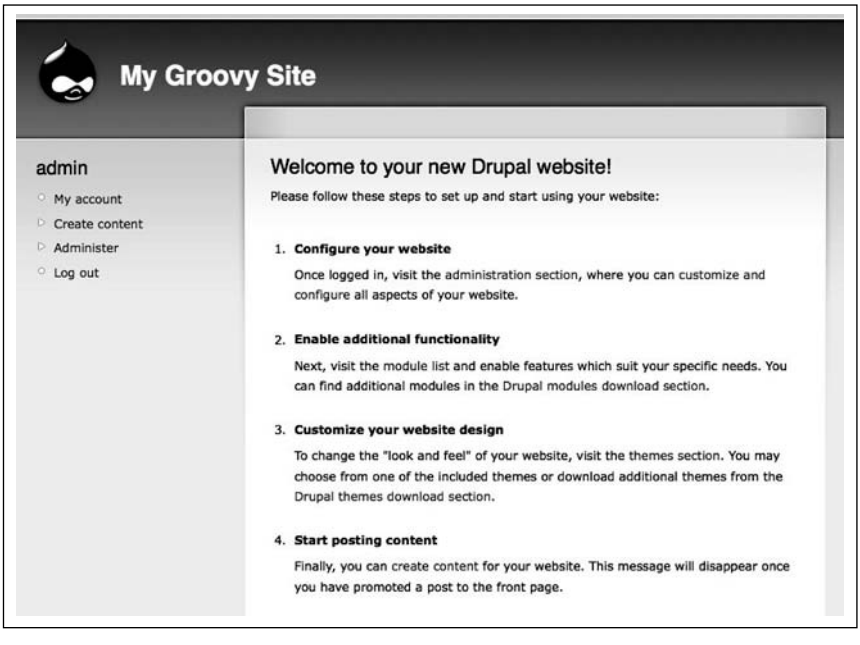


Рис. А.8. *Вновь установленный сайт на базе Drupal*

Обновление Drupal

Недостаточно просто установить Drupal, вам также потребуется своевременно обновлять его. Время от времени выходят новые версии модулей и ядра Drupal – чаще с исправлениями ошибок, иногда с реализацией дополнительных возможностей, а иногда с исправлениями критических проблем безопасности.

Нумерация версий

При обсуждении вопросов обновления системы совсем не лишним будет иметь некоторое представление о порядке нумерации версий Drupal. Самые полные сведения по этой теме вы найдете на странице *http://drupal.org/handbook/version-info*, а в общих чертах порядок нумерации представлен на рис. А.9.

Каждый выпуск ядра Drupal «основной» версии получает новый номер: Drupal 5, Drupal 6, Drupal 7 и так далее. Новые основные версии Drupal

выпускаются каждые 12–18 месяцев, они реализуют новые возможности, обладают улучшенными эксплуатационными характеристиками и более гибким прикладным интерфейсом. На протяжении всего срока жизни основной версии выпускается несколько «дополнительных» версий Drupal, таких как 6.0, 6.1 или 6.2. Дополнительные версии Drupal исправляют критические проблемы безопасности и серьезные ошибки.

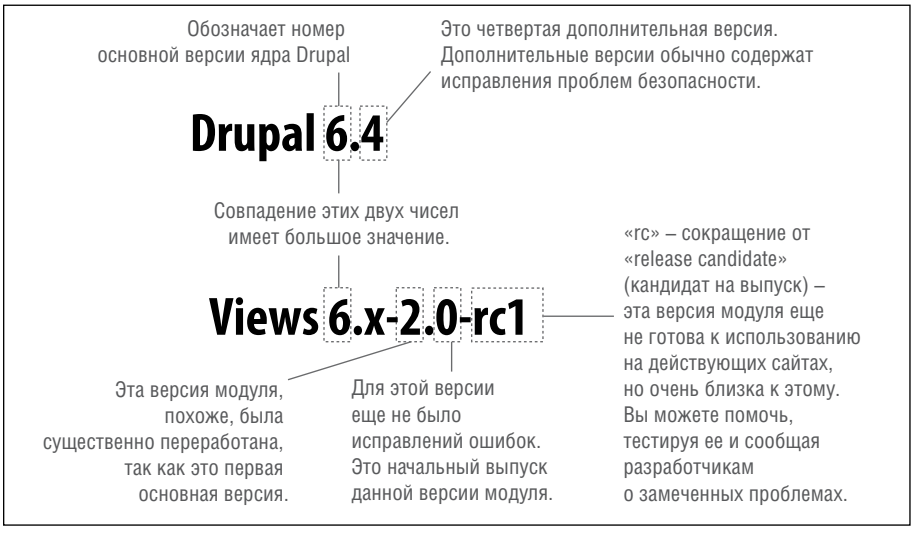


Рис. А.9. *Порядок нумерации версий Drupal*

Нумерация версий сторонних модулей, тем оформления и переводов производится в соответствии с такой схемой, как 6.x-1.3, где часть «6.x» обозначает основной номер версии Drupal, для которой предназначен этот модуль, тема или перевод – в данном случае Drupal 6. Число «1» обозначает «основной» номер версии *стороннего модуля*. А число «3» говорит о том, что это третий выпуск основной версии модуля с исправлениями ошибок.

Некоторые номера версий могут нести в себе «дополнительные» сведения о версии, например «-beta4» или «-rc2». В данном случае дополнения указывают, что модули пока находятся на этапе разработки, но их можно использовать для тестирования.

Обновление при переходе от одной дополнительной версии ядра Drupal или модуля к другой, например переход от версии Drupal 6.3 к версии 6.4 или от версии модуля Views 6.x-2.0 к версии 6.x-2.1 обычно проходит безболезненно, при условии, что обновления производятся регулярно. Но обновление при переходе от одной основной версии к другой, например от версии Drupal 6.3 к версии 7.0 или от версии модуля Organic Groups 6.x-1.0 к версии 6.x-2.0, и особенно к версии 7.x-1.0, требует особого внимания, так как обычно различия между версиями весьма существенны.

Об обратной совместимости

Проект Drupal придерживается такой политики сохранения обратной совместимости между основными версиями (такими как Drupal 6.x и 7.x), которая позволяет разработчикам свободно изменять *программный код*, лежащий в основе, но обязывает их обеспечивать возможность переноса пользовательских *данных*. Если разработчик обнаруживает более простой, более быстрый или более очевидный способ выполнения каких-либо операций, ему разрешается (и даже приветствуется) изменить программный код реализации. Это позволяет Drupal оставаться на передовых позициях и не тянуть за собой устаревший программный код, который требовалось бы поддерживать и сопровождать в течение многих лет. Однако в результате такой политики авторы сторонних модулей вынуждены следить за такими изменениями и вносить необходимые изменения в свои модули, чтобы обеспечить совместимость при обновлении до следующей основной версии.

Кроме того, в настоящее время проект Drupal придерживается политики поддержки только текущей и предыдущей версии. На момент написания этих строк самой свежей была основная версия Drupal 6, тем не менее продолжали выходить обновления и исправления ошибок для обеих версий, Drupal 6 и Drupal 5. Однако с выходом версии Drupal 7 поддержка Drupal 5 будет прекращена.

Для вас, как для пользователя Drupal, все вышеизложенное означает, что часто между выходом новой основной версии Drupal и появлением ключевых сторонних модулей для этой версии может пройти несколько месяцев. Кроме того, вы должны предусматривать переход сайта на использование новой основной версии Drupal каждые 18–24 месяца.

Более подробно с политикой сохранения обратной совместимости проекта Drupal можно ознакомиться на странице *http://drupal.org/node/65922*.

Модуль Update Status

В состав ядра Drupal 6 входит модуль с названием Update Status, который периодически проверяет наличие на сайте Drupal.org новых версий модулей, тем оформлений и самой системы Drupal. В случае появления новой версии одного из проектов или выхода нового выпуска с исправлениями безопасности на всех страницах административного раздела будет выводиться предупреждение красным цветом, предлагающее перейти на страницу Administer (Администрирование)→Reports (Отчеты)→Available updates (Доступные обновления) (*admin/reports/updates*) за дополнительной информацией. Вы можете также подписаться на рас-

сылку извещений об исправлениях в системе безопасности, по адресу *http://drupal.org/security*, и/или на ленту новостей Security RSS (*http://drupal.org/security/rss.xml*).

На странице Available updates (Доступные обновления), как показано на рис. А.10, отображается список проектов, установленных на вашем веб-сайте, отмеченных цветом в соответствии с состоянием доступных обновлений.

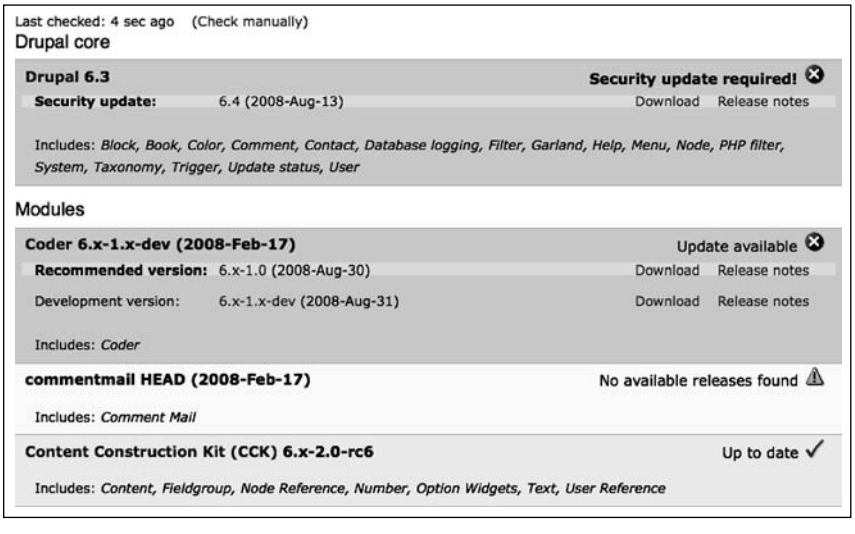


Рис. А.10. *Модуль Update Status показывает наличие обновлений для различных проектов*

Ниже описывается, каким цветом обозначаются различные состояния:

Красный

Новая версия проекта, рекомендуемая к установке, и версия проекта, установленная на веб-сайте. Особое внимание обращайте на проекты, отмеченные примечанием «Security update required!» (Требуется обновление системы безопасности!), и немедленно загружайте и устанавливайте рекомендованную версию.

Желтый

Модуль Update Status не в состоянии определить состояние проекта. Такое может случаться с такими проектами, как ваши собственные модули и темы оформления, или в случае, когда проект был получен не с сайта Drupal.org, или в случае проблем, возникших при получении информации о состоянии проекта.

Зеленый

Установлена самая свежая версия проекта. Никаких дополнительных действий не требуется.

Модуль Update Status может производить очень много «шума», особенно когда установлено достаточно большое число модулей – в течение недели несколько проектов модулей могут объявлять о выходе новых версий, если они продолжают активно развиваться. Вы можете отрегулировать вывод извещений, установив определенный порог на вкладке Settings (Настройки), на странице Administer (Администрирование)→Reports (Отчеты)→Available updates (Доступные обновления) (*admin/reports/updates/settings*), чтобы, например, извещения появлялись только в случае появления выхода исправлений в системе безопасности, обязательных к установке, а не обычных исправлений ошибок.

К обновлениям системы безопасности следует относиться очень серьезно и производить обновление как можно быстрее. За дополнительной информацией об исправлениях и нововведениях, предлагаемых обновлением, обращайтесь к примечаниям к выпускам модулей.

Кроме того, существует сторонний модуль Upgrade Status (*http://drupal.org/project/upgrade_status*), похожий на модуль Update Status, который отображает похожую информацию об установленных модулях и о наличии их реализаций для следующей основной версии Drupal. Такая возможность может пригодиться, когда планируется переход на новую основную версию, например переход от Drupal 6 к Drupal 7.

Режим профилактики сайта

На странице Administer (Администрирование)→Site configuration (Настройка сайта)→Site maintenance (Сайт на обслуживании) (*admin/settings/site-maintenance*), изображенной на рис. A.11, имеется возможность «отключить» сайт перед выполнением обновлений. Этот режим удобно использовать, когда установка обновлений может вызвать появление временных ошибок в работе. В режиме профилактики сайт становится недоступен для обычных пользователей, но не для администраторов. Бывает нежелательно позволять пользователям создавать новые материалы, когда вы производите обновление базы данных, так как это может привести к потере данных или к отображению сообщений об ошибках перед посетителями сайта. Кроме того, вы также можете определить текст сообщения, которое будет отображаться перед посетителями при отключении сайта, как показано на рис. A.11.

Если вам необходимо иметь возможность регистрироваться на сайте, когда он находится в режиме профилактики, вашей учетной записи должна быть присвоена роль, обладающая разрешением «administer site configuration» (управлять настройками сайта). Чтобы попасть в форму регистрации, откройте страницу *http://www.example.com/user*.

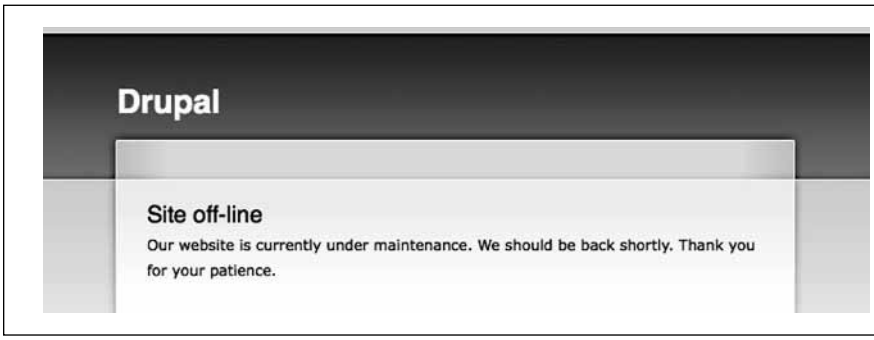


Рис. А.11. *Веб-сайт на базе Drupal находится в режиме профилактики*

Сценарий update.php

Сценарий *update.php*, работа которого изображена на рис. А.12, автоматически выполняет любые изменения в базе данных, необходимые для перехода от одной версии модуля к другой. Будь то переход от одной дополнительной версии к другой или переход к другой основной версии Drupal или стороннего модуля, сценарий *update.php* обеспечит корректный перенос данных.

Сценарий проверит все включенные модули на сайте и определит, какие изменения требуется выполнить. Индикатор хода выполнения операции показывает количество обновляемых модулей. По завершении он создает отчет об изменениях, произведенных в базе данных, и о любых ошибках, возникших в процессе обновления.

Сценарий *update.php* вносит изменения в базу данных, поэтому *очень важно* создавать резервную копию базы данных перед запуском этого сценария. Все необходимые инструкции по созданию резервных копий вы найдете в справочном руководстве Drupal по адресу *http://drupal.org/upgrade/backing-up-the-db*.

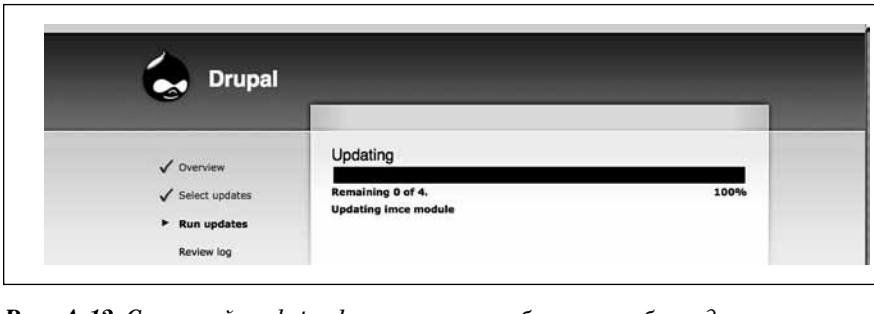


Рис. А.12. *Сценарий update.php выполняет обновление базы данных при переходе к другой версии*

Сценарий *update.php* должен запускаться только пользователем №1. Если вы не используете учетную запись пользователя №1, необходимо отредактировать файл *settings.php*, чтобы получить возможность запускать сценарий обновления. Вы должны изменить переменную `$update_free_access` в файле *settings.php* так, чтобы она имела значение не FALSE, а TRUE.

Будьте осторожны, если вы изменили значение этой переменной в файле *settings.php*, *обязательно* верните значение FALSE сразу же после запуска сценария обновления! В противном случае анонимные пользователи получат возможность запустить повторное обновление базы данных, что может вызвать появление самых разных проблем.

Обновление ядра Drupal

Процедура обновления сайта часто кажется более страшной, чем есть на самом деле. Помимо файла *UPGRADE.txt*, входящего в состав ядра Drupal, отличное описание процедуры обновления можно найти в справочном руководстве по адресу *http://drupal.org/upgrade/*, а также получить помощь на форуме поддержки *http://drupal.org/forum/21*. Самый важный шаг, который не следует забывать, – это создание и проверка резервных копий сайта.

Никогда не лишне еще раз подчеркнуть, насколько важно создавать резервные копии перед выполнением обновлений. Это в равной степени относится и к обновлению ядра Drupal, и к обновлению сторонних модулей. Обязательно создавайте резервные копии обеих составляющих сайта на базе Drupal: дерева каталогов и базы данных. В каждой системе может быть свой способ создания резервных копий, о каждом из которых невозможно рассказать в рамках этого приложения. Со своими вопросами вы можете обратиться к своему системному администратору или поискать ответы в разделе, посвященном созданию резервных копий, в руководстве по обновлению на сайте Drupal.org, по адресу *http://drupal.org/upgrade/backing-up-the-db*. Не забывайте тестировать резервные копии, чтобы быть уверенными, что вы всегда сможете восстановить свой сайт, если что-то пойдет не так.

Для обновления устаревшего модуля необходимо вручную загрузить необходимые файлы с сайта Drupal.org и переписать их на сервер, точно так же, как это делалось при первоначальной установке модуля. Система Drupal не загружает обновления автоматически. Это сделано с целью предотвратить затирание существующего кода модуля до того, как вы получите возможность протестировать его. Например, вполне возможно, что изменения в модуле могут потребовать обновить версию PHP, что могло бы вызывать фатальные ошибки, если бы обновления загружались вслепую. Всегда тестируйте обновленные модули на испытательном сервере, прежде чем развертывать их на действующем сайте.

В этом разделе описываются шаги по обновлению ядра Drupal в пределах основной версии, до следующей дополнительной версии, например при обновлении версии Drupal 6.3 до версии Drupal 6.4. При обновлении до новой основной версии Drupal, такой как Drupal 7, выполняются практически те же самые шаги, за исключением того, что в последнем случае необходимо одновременно выполнить обновление всех сторонних модулей и тем оформления.

1. Загрузите последний выпуск вашей версии Drupal, следуя инструкциям в разделе «Загрузка Drupal», приведенным выше.
2. Прежде чем что-либо сделать, создайте резервные копии базы данных и файлов. Следуйте рекомендациям вашего системного администратора или руководства по созданию резервных копий, по адресу *http://drupal.org/upgrade/backing-up-the-db*.
3. После создания резервных копий зарегистрируйтесь на сайте с учетной записью пользователя №1.
4. Перейдите на страницу Administer (Администрирование)→Site configuration (Настройка сайта)→Site maintenance (Сайт на обслуживании) (*admin/settings/site-maintenance*) и выберите радиокнопку Offline (Обслуживание). Вы можете отредактировать текст сообщения в поле Site offline message (Сообщение о перерыве в работе сайта). Щелкните на кнопке Save configuration (Сохранить конфигурацию), чтобы перевести сайт в режим профилактики.

Если сайт находится в режиме профилактики и вы вышли до того, как профилактика закончилась, вы по-прежнему сможете зарегистрироваться на сайте, перейдя на страницу регистрации, для чего нужно ввести в адресную строку броузера адрес *http://example.com/user*.

5. При обновлении до новой основной версии также рекомендуется перейти на страницу Administer (Администрирование)→Site building (Конструкция сайта)→Themes (Темы оформления) (*admin/build/themes*) и выбрать тему, входящую в состав ядра Drupal, например Garland или Bluemarine. Это поможет предотвратить появление ошибок в случае, если изменения коснулись элементов системы, которые используются текущей темой оформления.
6. Извлеките файлы из тарболла и замените новыми файлами все существующие файлы на сервере.
7. Убедитесь, что все файлы вашего сайта присутствуют в резервной копии. Все сторонние и ваши собственные разработки, а также каталог *files*, должны присутствовать в каталоге *sites* в резервной копии. Извлеките копию каталога *sites* из вашей резервной копии и добавьте его в дерево каталогов Drupal. Если вы выполняли модификацию таких системных файлов, как *.htaccess* или *robots.txt*, восстановите их из резервной копии.

8. Теперь, когда все файлы на месте, можно выполнить обновление базы данных. Откройте в своем броузере страницу *http://example.com/update.php*. Перед вами откроется страница мастера обновления, где будут перечислены шаги, которые необходимо выполнить. Щелкните на кнопке Continue (Продолжить).
9. Перед вами откроется страница обновления. Щелкните на кнопке Update (Обновить), чтобы запустить сценарий.

Если развернуть группу параметров Select versions (Выбор версий), можно увидеть, обновление каких модулей будет выполнено. Модули, для которых имеются обновления, будут обозначаться номерами версий, такими как 6001, уже отмеченными в раскрывающихся списках. Система Drupal следит за номерами версий автоматически, поэтому вам не следует изменять их. Для модулей, не имеющих обновлений, будет отмечен пункт No updates available (Нет доступных обновлений). Даже если на этой странице не будет указано ни одного имеющегося обновления, вам все равно следует запустить сценарий *update.php*, так как он выполнит очистку кэша и гарантирует, что система Drupal обнаружит все новые файлы. Отказ от запуска сценария может вызвать необъяснимое поведение обновленного сайта, пока кэш не будет очищен на странице Administer (Администрирование)→Site configuration (Настройка сайта)→Performance (Производительность) (*admin/settings/performance*).

10. По окончании работы сценария откроется страница, сообщающая о завершении операции обновления. Если перед обновлением вы изменяли тему оформления сайта, включите тему, использовавшуюся ранее, на странице Administer (Администрирование)→Site building (Конструкция сайта)→Themes (Темы оформления) (*admin/build/themes*).
11. Походите по страницам сайта и убедитесь, что обновление прошло успешно. Как только вы убедитесь, что все в порядке, вернитесь на страницу Administer (Администрирование)→Site configuration (Настройка сайта)→Site maintenance (Сайт на обслуживании) (*admin/settings/site-maintenance*), выберите радиокнопку Online (Работа) и щелкните на кнопке Save configuration (Сохранить конфигурацию), чтобы опять включить сайт.

Обновление сторонних модулей

Сторонние модули и темы оформления для Drupal обычно развиваются быстрее, чем ядро системы, и поэтому обновлять их приходится чаще. Вы можете обновлять сразу несколько модулей, хотя лучше всего обновлять их по одному, чтобы уменьшить вероятность появления ошибок и упростить поиск проблем, которые могут возникнуть в ходе обновления.

Ниже перечислены шаги, которые следует выполнять при обновлении сторонних модулей. Вы без труда заметите, что процесс обновления сторонних модулей напоминает процесс обновления ядра Drupal, отличие заключается только в файлах, которые изменяются.

1. Так же как и при обновлении ядра Drupal, вам необходимо загрузить последнюю версию модуля, который требуется обновить. Всегда просматривайте примечания к выпуску для модуля, чтобы убедиться, что там отсутствуют дополнительные инструкции по обновлению.
2. Даже если обновляется единственный модуль, по-прежнему важно создать резервные копии всей системы Drupal. Если что-то пойдет не так, у вас сохранится возможность восстановить сайт до состояния, предшествовавшего обновлению. Поэтому, прежде чем продолжить, обязательно создайте резервные копии.
3. Зарегистрируйтесь на сайте с учетной записью пользователя №1.
4. Перейдите на страницу Administer (Администрирование)→Site configuration (Настройка сайта)→Site maintenance (Сайт на обслуживании) (*admin/settings/site-maintenance*) и выберите радиокнопку Offline (Обслуживание). Вы можете отредактировать текст сообщения в поле Site offline message (Сообщение о перерыве в работе сайта). Щелкните на кнопке Save configuration (Сохранить конфигурацию), чтобы перевести сайт в режим профилактики.
5. Теперь можно заменить файлы модуля. Извлеките файлы из архива, загруженного с сайта Drupal.org, и замените ими прежние файлы модуля. Для большей надежности можно сначала полностью удалить папку с прежней версией модуля и затем добавить папку с новой версией в каталог *sites/all/modules*.
6. Перейдите на страницу *http://example.com/update.php* и щелкните на кнопке Continue (Продолжить).
7. Щелкните на кнопке Update (Обновить), чтобы запустить сценарий обновления.
8. Вернитесь на сайт и посмотрите, все ли в порядке. Особое внимание следует обратить на функциональность, которая обеспечивается модулем или модулями, чтобы убедиться в отсутствии ошибок.
9. Повторите шаги с 5 по 8 для каждого обновляемого модуля.
10. По завершении вернитесь на страницу Administer (Администрирование)→Site configuration (Настройка сайта)→Site maintenance (Сайт на обслуживании) (*admin/settings/site-maintenance*), выберите радиокнопку Online (Работа) и щелкните на кнопке Save configuration (Сохранить конфигурацию), чтобы включить сайт.

Ссылки

Ниже приводится перечень модулей, упоминавшихся в этом приложении:

- Update Status: ядро
- Upgrade Status: *http://drupal.org/project/upgrade_status*

Ниже приводится перечень внешних ссылок, использовавшихся в этом приложении:

- Веб-сервер Apache: *http://httpd.apache.org*
- Пакет MAMP: *http://mamp.info/en/download.html*
- СУБД MySQL: *http://mysql.com*
- PHP: *http://php.net*
- СУБД PostGreSQL: *http://postgresql.org*
- Пакет WampServer: *http://www.wampserver.com*
- Пакет XAMPP: *http://www.apachefriends.org/en/xampp.html*

Ниже приводится перечень ресурсов на сайте Drupal.org, упоминавшихся в этом приложении:

- Проект Drupal: *http://drupal.org/project/drupal*
- Об обратной совместимости: *http://drupal.org/node/65922*
- О создании резервных копий файлов и базы данных[1]: *http://drupal.org/upgrade/backing-up-the-db*
- Чистые ссылки (Clean URLs): *http://drupal.org/node/15365*
- Руководство по установке[2]: *http://drupal.org/getting-started/6/install*
- Создание многосайтовой установки: *http://drupal.org/node/43816*
- Безопасность: *http://drupal.org/security*
- Лента RSS о безопасности: *http://drupal.org/security/rss.xml*
- Системные требования: *http://drupal.org/requirements*
- Руководство по обновлению: *http://drupal.org/upgrade*
- Форум по обновлению Drupal: *http://drupal.org/forum/21*

---

[1] Похожий материал на русском языке: *http://www.drupal.ru/node/13258*. – *Примеч. перев.*

[2] Похожий материал на русском языке: *http://www.cmsmagazine.ru/library/items/cms/drupal_intro/*. – *Примеч. перев.*

B

Выбор модулей

Когда существует более 2000 модулей, доступных для выбора, и каждый день добавляются новые, поиск сторонних модулей, необходимых для решения определенных задач, превращается в сложный процесс. На протяжении всей книги авторы старались подсказать названия модулей, которые «должен иметь каждый», особенно выделяя архитектурные модули, которые часто используются при конструировании веб-сайтов на базе Drupal. Мы также стремились охватить модули, уже зарекомендовавшие себя с положительной стороны, и которые, скорее всего, будут продолжать использоваться при создании сайтов.

Однако каждый новый проект веб-сайта имеет свои уникальные требования, которые могут выходить за рамки этой книги, и к тому же состав репозитория сторонних модулей постоянно изменяется. Модули, которые когда-то были важными строительными блоками, могут оказаться заброшенными или вытесненными превосходящими их вариантами. Могут появляться новые модули, которые полностью затмевают то, что существовало до них.

В этом приложении мы попытаемся рассказать о некоторых приемах, которыми пользуются «знающие» люди для оценки и выбора модулей. Следует понимать, что не существует простого набора рекомендаций (включая представленный здесь), которые помогли бы вам получить полную информацию о модуле. Важно помнить, что тщательность в оценке при выборе модулей поможет в будущем избежать неприятных сюрпризов.

Поиск модулей

Первый шаг при выборе модуля, отвечающего вашим потребностям, заключается в том, чтобы фактически *отыскать* его. К счастью, все модули Drupal (за несколькими редкими исключениями) находятся непо-

средственно на основном веб-сайте проекта Drupal.org, поэтому при поиске модулей можно пользоваться единственным ресурсом. Ниже рассказывается, как это делается.

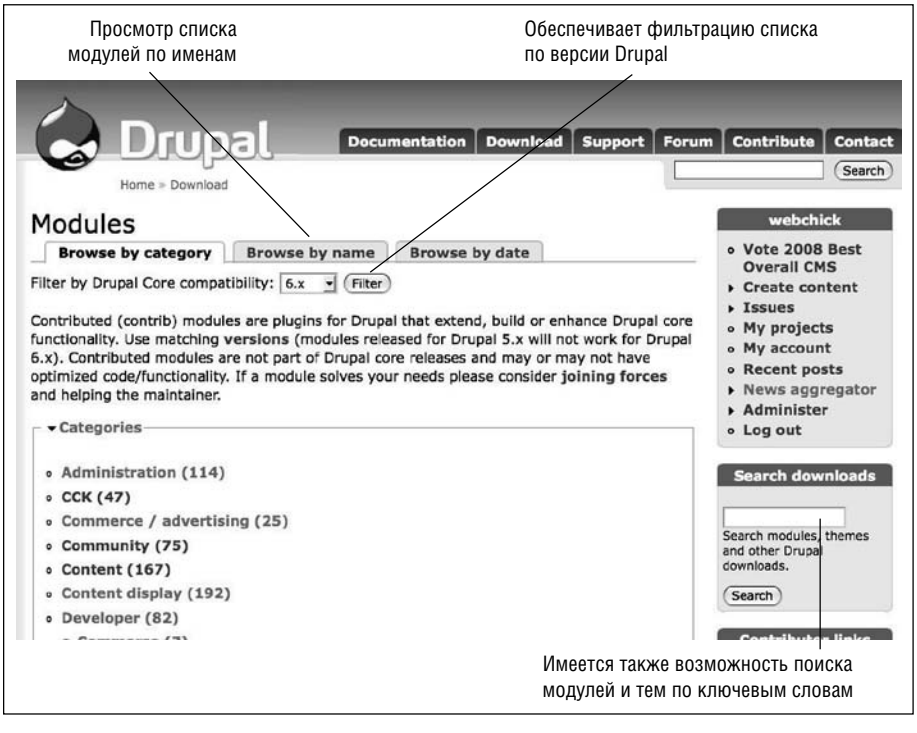


Рис. В.1. *Страницы просмотра репозитория модулей на сайте Drupal.org*

Просмотр списков модулей

Страница со списками модулей находится по адресу *http://drupal.org/project/Modules*, изображенная на рис. В.1, перечисляет модули по категориям (таким как CCK или электронная почта), по именам в алфавитном порядке и по дате последнего обновления. Просмотр этой страницы может помочь отыскать модули, предназначенные для решения определенных задач, или модули, которые часто обновляются и продолжают активно развиваться.

Модули для версии Drupal 5.x несовместимы с версией Drupal 6.x и наоборот. Чтобы оставить в списках только те модули, которые можно использовать на вашем сайте, измените значение фильтра Filter by Drupal Core compatibility (Фильтровать по совместимости с ядром Drupal) в соответствии с используемой версией Drupal. Доступ к этому фильтру вы получите только после регистрации на веб-сайте Drupal.org. Учетная запись создается

просто и бесплатно, причем ее наличие откроет вам доступ ко многим полезным инструментам.

Еще одна полезная особенность сайта Drupal.org, которая позволит отказаться от подписки на ленту RSS, – это страница *http://drupal.org/taxonomy/term/14*, отображаюшая самые новые модули по мере их создания.

Поиск по ключевым словам

Кроме того, на сайте Drupal.org имеется блок для поиска и загрузки модулей, также изображенный на рис. В.1. Например, поиск по слову «wiki» вернет список модулей, в названиях или описаниях которых содержится это ключевое слово. Это позволяет сузить область поиска модулей в соответствии со своими потребностями и ускорить поиск, в сравнении с просмотром списков по категориям.

Форумы Drupal.org

На сайте Drupal.org имеются форумы поддержки по адресу *http://drupal.org/forum*. В частности, там имеется ветка «Before you start» (Прежде чем начать) по адресу *http://drupal.org/forum/20*, где вы сможете получить массу информации в форме ответов на вопросы других пользователей, касающиеся использования модулей. На форуме вы нередко будете обнаруживать полезные советы не только по тем особенностям, которые вы пытаетесь реализовать в настоящее время, но и по проблемам, с которыми вам еще предстоит столкнуться. В ветке «Drupal showcase» (Витрина Drupal), по адресу *http://drupal.org/forum/25*, вы найдете немало людей, с гордостью рассказывающих о своих веб-сайтах, построенных на базе Drupal, – они будут счастливы поделиться с вами информацией о том, как были реализованы те или иные особенности.

Примеры

Достаточно велика вероятность того, что, какой бы уникальной ни казалась вам ваша задача, с аналогичной ранее уже кто-то сталкивался и успешно решил ее. Вы можете существенно сократить время, затрачиваемое на поиск модуля, узнав, как другие решили эту проблему. Справочное руководство Drupal содержит раздел с примерами по адресу *http://drupal.org/cases*. Здесь вы найдете примеры, порой достаточно крупных веб-сайтов, использующих систему Drupal, и описание, почему была выбрана именно система Drupal и как был сконструирован тот или иной сайт. В число наиболее представительных примеров входят:

- Журнал «Popular Science»: *http://drupal.org/node/233090*
- Sony BMG MyPlay: *http://drupal.org/node/241344*
- New York Observer: *http://drupal.org/node/141187*

Планета Drupal

Страница «Планета Drupal» (*http://drupal.org/planet*), изображенная на рис. B.2, – это собрание блогов членов сообщества Drupal и одновременно отличная возможность получения свежих новостей, касающихся модулей. Здесь часто можно встретить ссылки на различные руководства по использованию модулей, обзоры и новости, а кроме того, «Планета Drupal» позволит вам постоянно держать руку на пульсе событий, происходящих в сообществе.

Рис. B.2. *Страница «Планета Drupal», где собирается информация из блогов членов сообщества Drupal и компаний*

Сторонние веб-сайты

На странице *http://drupal.org/node/289913* имеется список сторонних веб-сайтов, то есть сайтов, независимых от Drupal.org, где часто можно найти информацию, полезную для оценки модулей. Например, на сайте *http://drupalmodules.com* приводятся оценки пользователей и обзоры модулей Drupal, а на сайте *http://www.lullabot.com* имеется множество различных статей, видео- и аудиоматериалов, многие из которых рассказывают о популярных модулях и о том, как ими пользоваться.

Оценка состояния модуля

Устойчивость любого открытого проекта в первую очередь зависит от способностей основных его разработчиков, и проект Drupal в данном случае не является исключением. Даже при том, что каждая строчка программного кода, добавляемая в ядро Drupal, проходит строгую экс-

пертизу, сторонние модули больше напоминают «дикий запад», где любой преодолевший несколько начальных препятствий считает для себя возможным предлагать всем желающим модули для загрузки. Сообщество Drupal стремится устанавливать как можно меньше барьеров для тех, кто желает поделиться своими разработками, чтобы стимулировать рост и процветание сообщества разработчиков Drupal. Такой подход имеет свои достоинства (для практически любой проблемы имеется свой модуль, решающий ее полностью или частично) и недостатки (уровень мастерства разработчиков варьирует в весьма широких пределах, поэтому в программном коде могут присутствовать проблемы, связанные с эффективностью или безопасностью, а сами разработчики могут оказаться загружены другими делами и оказаться не в состоянии обеспечить сопровождение своих модулей на должном уровне).

Уровень поддержки модуля, качество программного кода и широта использования в сообществе – все это очень важные факторы, которые следует учитывать при выборе модулей. В данном разделе рассказывается о том, как оценить эти факторы с помощью инструментов, предоставляемых сайтом Drupal.org, начиная с самой важной особенности любого модуля Drupal: со страницы проекта.

Страница проекта

Модули, темы оформления, переводы и даже само ядро Drupal – все это *проекты* на сайте Drupal.org. Каждый проект имеет собственную страницу по адресу *http://drupal.org/project/file_name*, где содержится информация, позволяющая оценить состояние модуля.

На рис. В.3 показана первая часть страницы проекта модуля. Здесь можно узнать имя человека, *осуществляющего поддержку модуля* (обычно это сам автор и/или один из основных разработчиков), дату создания проекта, описание назначения модуля и иногда снимки с экрана, демонстрирующие модуль в действии.

Дата создания проекта может пригодиться при поиске решений, проверенных временем (если модуль был создан неделю тому назад, вероятно лучше будет дать ему возможность немного созреть, прежде чем обрести зависимость от него). Однако следует также знать, что некоторые давнишние модули могут считаться устаревшими решениями и вытесняться более современными модулями.

Еще ниже находится таблица со списком выпущенных версий модуля (как показано на рис. В.4), о которой коротко рассказывалось в главе 2. В ней заключено множество полезной информации, такой как дата последнего обновления; наличие «официальных версий» модуля, которые являются стабильными версиями; ссылки на примечания к выпускам для каждой версии, где описывается, какие ошибки были устранены и какие новые возможности были добавлены; и ссылка на список со всеми версиями модуля, даже устаревшими.

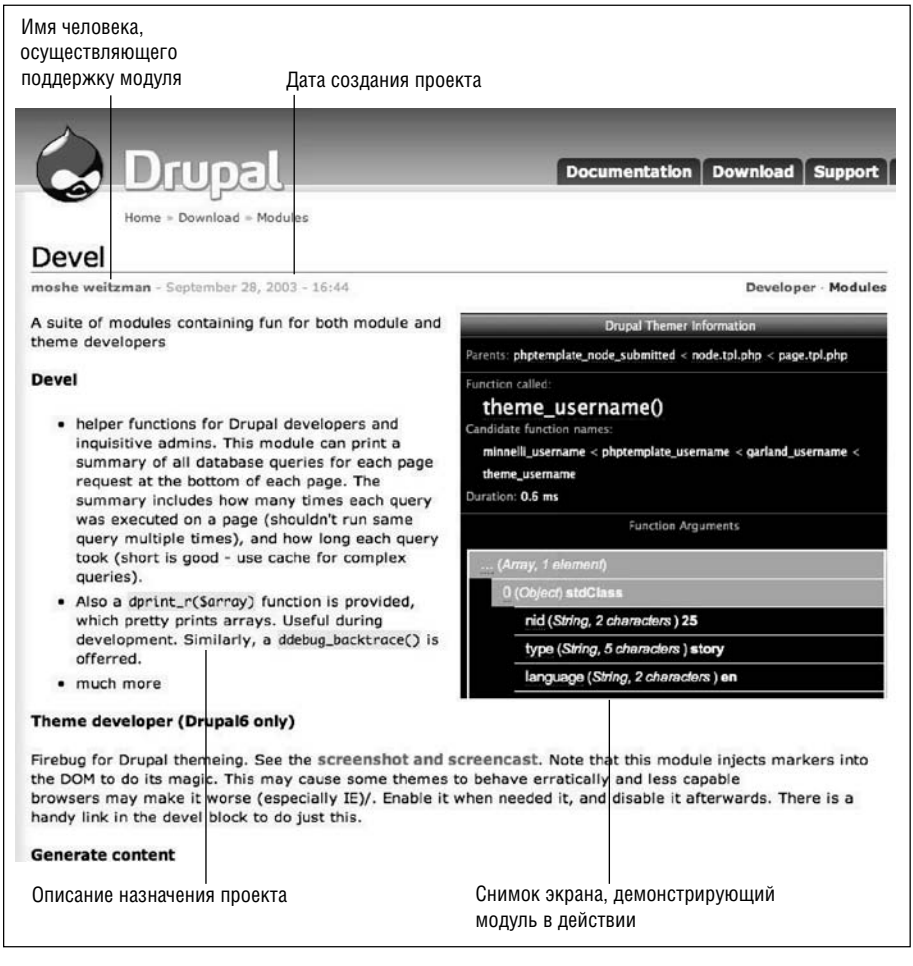


Рис. В.3. *Страница проекта модуля Devel*

Данная таблица со списком выпущенных версий, взятая со страницы проекта модуля Date 7 сентября 2008 года, явно демонстрирует здоровое состояние проекта. У модуля Date имеются стабильные версии, как для Drupal 5, так и для Drupal 6 (версия для Drupal 6 является всего лишь кандидатом на выпуск, тем не менее она близка к завершению и доступна для тестирования). Дата выпуска разрабатываемых версий свидетельствует, что программный код был обновлен совсем недавно, а это означает, что проект активно развивается. Щелкнув на ссылке View all releases (Просмотреть все выпуски), вы получите список версий модуля, где можно будет увидеть версии для Drupal 4.7 и даже для разрабатываемой версии Drupal 7. Несмотря на то что версия Drupal 7 в настоящее время активно разрабатывается, тем не менее выпущена она будет еще не скоро.

Releases

Official releases	Date	Size	Links	Status
6.x-2.0-rc2	2008-Aug-21	184.3 KB	Download · Release notes	Recommended for *6.x* ✓
5.x-2.1	2008-Sep-05	174.38 KB	Download · Release notes	Recommended for *5.x* ✓
5.x-1.8	2007-Dec-28	98.21 KB	Download · Release notes	Supported for *5.x* ⚠

Development snapshots	Date	Size	Links	Status
6.x-2.x-dev	2008-Sep-06	205.07 KB	Download · Release notes	Development snapshot ⊗
5.x-2.x-dev	2008-Sep-06	186.12 KB	Download · Release notes	Development snapshot ⊗

• View all releases

Рис. В.4. *Таблица со списком выпущенных версий модуля*

С другой стороны, следующие признаки должны вызывать настороженность:

- Если проект имеет только разрабатываемые версии и ни одного официального выпуска или если у него на странице *вообще отсутствует* таблица со списком версий, это свидетельствует о том, что проект пока находится в стадии разработки и пока не готов к использованию на действующих веб-сайтах.
- Если с момента последнего обновления прошло несколько месяцев, это может свидетельствовать об отсутствии у разработчиков активности и интереса к модулю. Это может также означать, что вы нашли абсолютно завершенный модуль, в котором отсутствуют ошибки и который не нуждается в дальнейшем усовершенствовании, но это крайне редкий случай.

Наконец, внизу страницы проекта может находиться список ссылок на такие внешние ресурсы, как домашняя страница проекта на внешнем сайте, ссылка на документацию или на демонстрационный сайт. Наличие этих ссылок обычно говорит, что человек, осуществляющий сопровождение, влюблен в свой модуль и хотел бы повысить его качество, насколько это возможно.

Кроме того, не упускайте из виду статистику использования проекта – важнейший параметр при оценке популярности модуля в сравнении с другими.

Очереди вопросов

Разработка программного кода проектов в сообществе Drupal ведется с использованием *очередей вопросов*, пример которой со страницы *http://drupal.org/project/issues/3060* приведен на рис. В.5. Очередь вопросов служит местом регистрации ошибок, запросов на реализацию новых особенностей, просьб о поддержке и задач данного проекта. Эта очередь используется разработчиками проекта, как общее рабочее про-

странство. Любой член сообщества может зарегистрировать свой вопрос к участникам проекта, и любой может предложить свои решения.

Слово «fixed» (решено) означает, что кто-то из разработчиков проекта уже уделил внимание указанной проблеме

Если время, прошедшее с момента последнего обновления, находится в пределах одной-двух недель, это может служить признаком хорошего здоровья проекта. В данном случае видно, что с момента последнего обновления прошло не более двух часов!

Issues for *Drupal*

Create · Statistics · Subscribe · Advanced search

Project: Drupal | Status: active,active (needs mc | Category: <all> | Priority: <all> | Search

Summary	Status	Priority	Category	Version	Last updated	Assigned to
regarding to change my site status to online	fixed	normal	support request	5.10	8 min 37 sec	
D6 search refactoring (backport to 5)	patch (code needs review)	normal	task	5.x-dev	46 min 37 sec	douggreen
Remove time()	fixed	critical	bug report	7.x-dev	1 hour 14 min	Dave Reid
change E_NOTICE to warning	patch (code needs work)	normal	feature request	7.x-dev	1 hour 18 min	
Really destroy the session when logging out	active	normal	task	7.x-dev	1 hour 26 min	
menu_get_active_breadcrumb breaks breadcrumbs in p	patch (code needs review)	normal	bug report	7.x-dev	1 hour 44 min	
quotation marks trips max value	active	normal	bug report	6.3	2 hours 41 min	

Слово «patch» (заплата) говорит о том, что был написан программный код, исправляющий указанную ошибку. Обычно это свидетельствует о наличии активных разработчиков в сообществе, окружающем проект

Наличие имени в колонке «Assigned to» (Закреплено за) говорит о том, что кто-то уже занимается решением этой проблемы

Рис. B.5. *Очередь вопросов проекта ядра Drupal как пример здорового проекта*

Отыскать очередь вопросов проекта можно несколькими способами. Самый обычный из них заключается в том, чтобы просто перейти на страницу проекта и начать поиски оттуда. Если прокрутить страницу вниз, вы увидите список различных ссылок. Самая близкая ссылка в этом списке – это ссылка View open issues (Просмотр открытых проблем) в разделе Support (Поддержка). Вы можете также обратиться к общему списку всех проблем для всех проектов по адресу *http://drupal.org/project/issues*, и с помощью раскрывающегося списка Project (Проект) выбрать интересующий вас проект.

Поскольку очередь вопросов открывает доступ к информации о событиях, происходящих в ходе разработки проекта, возможность «читать» очередь является важным признаком здоровья проекта.

Можно было бы предположить, что проект с большим количеством проблем – это низкокачественный проект, а проект с небольшим числом проблем – это высококачественный проект. Несмотря на то что такая точка зрения имеет право на существование, тем не менее следует заметить, что сам проект Drupal в настоящее время имеет более 5000 открытых вопросов, хотя его программный код соответствует самым высоким стандартам качества. Гораздо чаще количество проблем в очереди указывает на высокую популярность проекта, а не на недостаточно высокое его качество.

С другой стороны, очень большое значение имеет само содержание заявленных вопросов. На рис. В.5 можно видеть ряд признаков, свидетельствующих о состоянии здоровья проекта. Здесь в списке присутствует пара вопросов, помеченных как решенные в течение последних 24 часов. Решение двух проблем было закреплено за разработчиками, что обычно говорит о том, что они взяли на себя ответственность за поиск решения. Для большинства проблем был выпущен программный код: в двух случаях – готовый для тестирования сообществом и в одном – требующий доработки, что по крайней мере свидетельствует о том, что проблема уже начала решаться. Все это показательно для здорового сообщества разработчиков, поддерживающего проект. Только две проблемы в списке помечены как «active» (нерешенные); это говорит о том, что они ждут, когда на них обратят внимание.

На рис. В.6 показана противоположная картина. Это очередь вопросов модуля Flexinode, который является предшественником современного модуля CCK. На первый взгляд картина напоминает очередь вопросов проекта Drupal, которую мы видели выше. Правда, здесь имеется немного больше «нерешенных» проблем и нет ни одной, которая была бы помечена как решенная. Однако здесь есть несколько проблем, для которых представлен программный код решения, а за некоторыми из них даже закреплены разработчики. Так что же тут не так?

Все дело в колонке Last updated (Последнее обновление), которая указывает, как давно был оставлен последний ответ на вопрос. В очереди вопросов проекта Drupal, как видно на рис. В.5, со времени получения последнего ответа прошел час-другой, а иногда и меньше – один из ответов был получен всего за восемь минут до запроса! Это означает, что помощь проекту оказывается постоянно людьми со всего мира. В случае же с проектом Flexinode самый последний ответ был получен 18 недель тому назад, а для большинства других вопросов – более года тому назад. Это явно говорит о том, что модуль заброшен и разработчики потеряли к нему интерес.

Большинство проектов модулей занимают промежуточное положение между этими двумя крайними точками. Есть проекты, содержащие во-

просы, которые давно не рассматривались, и, напротив, есть проекты, демонстрирующие более высокую активность. Выберите пару проблем и просмотрите на фактические ответы в них – это разработчики, сообщающие о том, что найдено решение, или другие отчаявшиеся пользователи, сообщающие: «У меня тоже наблюдается эта проблема. Есть ли решение?».

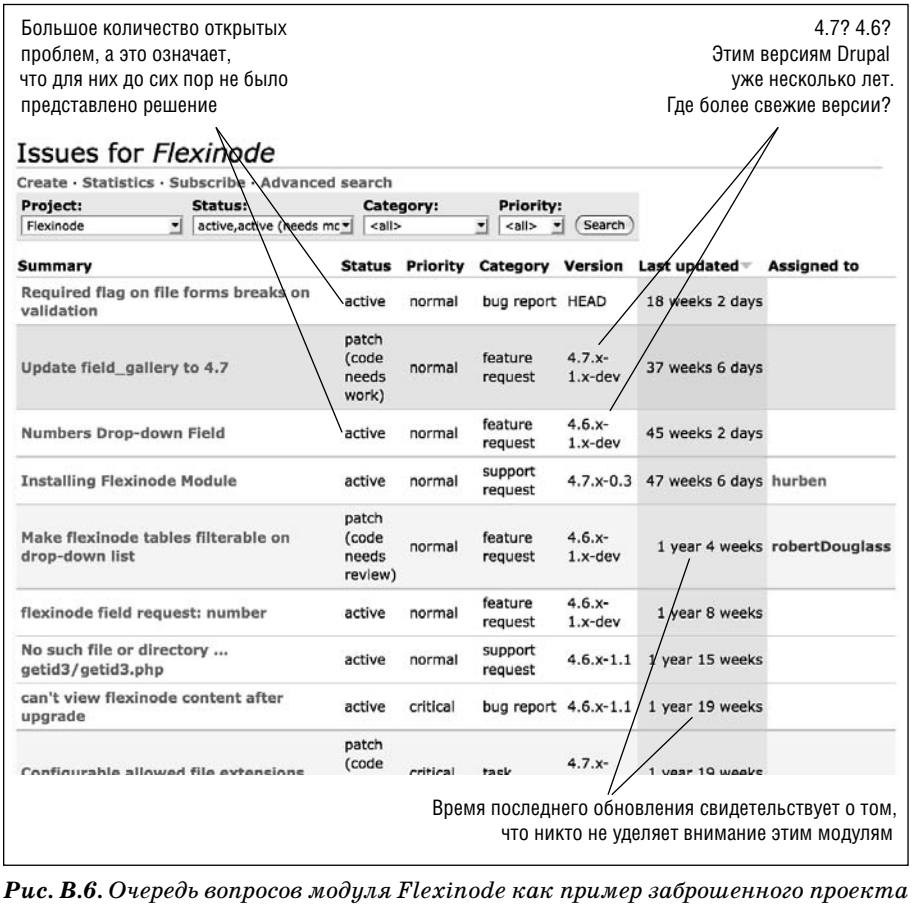


Рис. В.6. *Очередь вопросов модуля Flexinode как пример заброшенного проекта*

Программный код

Все сторонние модули для системы Drupal хранятся в центральном репозитории по адресу *http://cvs.drupal.org/viewvc.py/drupal/contributions/modules/*, куда можно заглянуть, чтобы получить представление о том, как выглядит реализация модуля, прежде чем загружать его. Очевидно, что те, кто знаком с языком PHP, смогут получить больше информации, но вообще говоря, любой в состоянии отличить качественный программный код. Ищите модули, содержащие ясный, хорошо документированный, хорошо организованный программный код, отвечаю-

щий стандартам оформления. Программный код, не соответствующий этим критериям, сложно сопровождать, и другим разработчикам будет сложнее подключиться к такому проекту.

Люди, стоящие за программным кодом

Каждый человек, помогающий проекту Drupal, – это уникальная личность, имеющая свой круг интересов, уровень подготовки и мотивацию к участию в проекте. Одни являются классными программистами, которые живут и дышат программным кодом. Другие нанимаются компаниями, занимающимися разработкой Drupal и предоставлением консультационных услуг, и получают плату за сопровождение своих модулей. Третьи – люди, увлеченные своим хобби, которые создают свои сайты и сопровождают один-два модуля, являющиеся основными для их сообществ. Есть и такие, кто оказывает помощь из удовольствия, просто потому, что им это нравится. Есть и такие, кто производит программный код, пока им это необходимо, но затем бросают его и переходят к другим, более интересным занятиям. И конечно, есть те, кто не попадает ни в одну из категорий, упомянутых выше.

Таким образом, при оценке модуля очень важно узнать побольше о тех, кто стоит за программным кодом. На сайте Drupal.org имеется несколько полезных особенностей, которые помогут вам в этом.

Активность разработчиков

Первое, на что стоит обратить внимание, – это ссылка Developers (Разработчики) внизу страницы проекта (например, *http://drupal.org/project/developers/3060*), щелкнув на которой, вы получите таблицу со списком разработчиков, как показано на рис. В.7, занимающихся (или занимавшихся) поддержкой этого проекта. В этой таблице отображается информация об изменениях, которые когда-либо *вносились* в проект всеми, кто имел такую возможность.

По этим сведениям можно получить общее представление о том, кто из разработчиков дольше всех участвует в данном проекте, насколько активно участвует каждый разработчик и насколько близко знаком с программным кодом проекта каждый из них. Признаком здорового, развивающегося проекта может служить большое число недавних изменений, переданных в проект, а также большое число участников в списке, если, конечно, в нем отсутствуют особо примечательные личности. Когда список слишком короткий, а последние изменения вносились несколько месяцев тому назад, особенно, когда исследование очереди вопросов вызывает настороженность, то, вероятно, стоит заняться поисками альтернативного решения или предложить разработчикам плату за необходимые вам изменения, чтобы снова возбудить у них интерес.

Developers for *Drupal*

User	Last commit	First commit	Commits
webchick	1 day ago	1 week ago	21 commits
Dries	1 day ago	8 years ago	16220 commits
drumm	1 week ago	2 years ago	1084 commits
Gábor Hojtsy	7 weeks ago	1 year ago	2390 commits
killes@www.drop.org	34 weeks ago	2 years ago	715 commits
Steven	1 year ago	7 years ago	2083 commits
Kjartan	1 year ago	8 years ago	694 commits
Natrak	4 years ago	8 years ago	174 commits
Jeroen	7 years ago	8 years ago	253 commits

Рис. В.7. *Список разработчиков проекта Drupal, где можно наблюдать активность разработки*

Профили пользователей

Всякий раз, когда на страницах сайта Drupal.org встречается имя пользователя, вы можете щелкнуть на нем, чтобы просмотреть профиль пользователя (например, *http://drupal.org/user/35821*), как показано на рис. В.8. Несмотря на то что здесь содержится информация, типичная для любого профиля пользователя на любом сайте, такая как имя и фамилия, список интересов, пол и страна проживания, тем не менее здесь имеется несколько дополнительных сведений, которые позволят больше узнать о человеке, стоящем за программным кодом.

Профиль пользователя начинается с краткой информации об участии пользователя в проекте Drupal. Здесь обычно упоминаются модули, написанные им, различные группы, в работе которых он принимает участие, такие как группа документирования или группа администраторов сайта, и другая подобная информация. Эти сведения помогут вам понять мотивацию данного человека и уровень его подготовки.

Эта информация сопровождается серией «примечаний», из которых можно узнать, оказывал ли пользователь помощь в работе над документацией, в поддержке других пользователей и в разработке модулей, а также – в каких конференциях, посвященных системе Drupal, он принимал участие. Каждое примечание является ссылкой, щелкнув на которой, можно получить список других пользователей, которые имеют такое же примечание. Пользователь с большим числом таких ссылок, как правило, более тесно связан с сообществом Drupal, чем другие пользователи, не имеющие таких ссылок.

Здесь вы найдете таблицу с описанием деятельности пользователя на сайте

Здесь вы найдете форму для передачи сообщения пользователю

quicksketch

View | Track | Contact

Drupal

Drupal contributions

I'm a Lullabot. I teach courses on theming, write articles, and generally try to make the world a more informed place about Drupal. My strengths are in detail oriented work, design, and jQuery. I'm most interested in user interface improvements and image handling.

* tabledrag.js (Drupal 6 drag and drop)
* AHAH Framework (Drupal 6)
* Fivestar Module (w/ Jeff Eaton)
* Link Module
* Webform updates, features and maintenance
* tinyMCE work (drupalbreak/drupalpagebreak plugin)
* Core patches
* Various module patching

Некоторые биографические сведения о пользователе, уровне его подготовки и круге интересов

I contributed Drupal modules
I help in the Drupal support forums
I provide Drupal-related services
I attended OSCMS 2007, Sunnyvale
I attended Drupalcon Barcelona 2007
I attended Drupalcon Boston 2008
I attended Drupalcon Szeged, Hungary 2008

Серия примечаний об участии в работе проекта, о заинтересованности в оплате, об участии в конференциях и т. д.

Рис. В.8. *Страница профиля пользователя на сайте Drupal.org*

На вкладках в верхней части страницы также можно найти немало полезной информации. На вкладке Track (История) отображается список всех сообщений, оставленных этим пользователем на сайте Drupal.org. Он позволит составить мнение о том, насколько тесно связан пользователь с сообществом Drupal, оценить степень его активности, а также узнать о его отношении к другим пользователям.

Вкладка Contact (Связаться), если пользователь разрешил ее, может использоваться для прямой связи с пользователем по электронной почте.

Несмотря на всю заманчивость идеи воспользоваться формой Contact (Связаться), чтобы отправить разработчику просьбу о поддержке или сообщение об ошибке, это считается дурным тоном. Время, которое разработчик потратит на ответ по электронной почте, будет *отнято* от времени, которое он мог бы потратить на дальнейшую разработку модуля или на помощь другим пользователям, испытывающим те же самые проблемы.

Чтобы сообщить о своей проблеме, всегда используйте очередь вопросов, так как это позволит дать ответ на ваш вопрос не только разработчику, но и другим, кто пользуется этим же модулем, и даст возможность другим пользователям найти этот ответ. Вообще, контактную форму разработчика следует использовать только для передачи частной информации, такой как предложение о найме на работу.

Контактная форма может также использоваться, чтобы вы могли выразить свою благодарность за хорошо проделанную работу, – большинство разработчиков модулей слышат от своих пользователей только жалобы о проблемах, поэтому любому из них будет приятно узнать, что кто-то остался доволен программным кодом, полученным бесплатно.

Еще ниже на странице имеется информация о том, как давно пользователь является членом сообщества Drupal.org, а также список проектов, в работе над которыми он принимал участие на протяжении этого времени, как показано на рис. B.9. У одних разработчиков в списке может быть указано один-два проекта, у других – более 50. Большое число проектов в списке обычно свидетельствует о том, что пользователь активно вовлечен одновременно во многое и, скорее всего, во многом хо-

History

Member for
2 years 47 weeks

Projects

Webform (1314 commits)
Fivestar (612 commits)
Link (185 commits)
Flag (112 commits)
Views Bookmark (76 commits)
External Links (67 commits)
ImageCache (39 commits)
ImageField (39 commits)
Install Profile API and Profile Wizard (25 commits)
Profile Privacy (25 commits)
Sandbox (22 commits)
Form single (11 commits)
Buddylist (8 commits)
MultiBlock (5 commits)
Survey-Webform Migrate (4 commits)
String Overrides (1 commit)
Node comments (1 commit)

Рис. B.9. *Внизу страницы профиля пользователя у каждого разработчика Drupal имеется список проектов, в работе над которыми он принимал участие*

рошо разбирается. С другой стороны, участие в *многочисленных* проектах может также свидетельствовать о том, что пользователь слишком разбрасывается и пытается делать слишком много всего одновременно, из-за чего может страдать качество его модулей.

Участие в проекте

Безусловно, самый эффективный способ постоянно находиться в курсе того, какие модули являются самыми полезными, и обеспечить соответствие этих модулей вашим требованиям заключается в том, чтобы принять непосредственное участие в их разработке. Сообщество Drupal предлагает огромное количество вариантов участия для *всех желающих*, начиная от тех, кто только вчера впервые установил Drupal, до тех, кто занимается программированием с пеленок.

Руководство «Getting Involved», находящееся по адресу *http://drupal.org/getting-involved*, расскажет вам обо всех вариантах участия в проекте Drupal. Ниже приводятся некоторые из вариантов, которые подойдут даже для тех, кто не занимается программированием:

Приведение в порядок очереди вопросов

При составлении своего мнения о модулях вы, так или иначе, окажетесь на странице с очередью вопросов. Почему бы вам не потратить несколько минут, чтобы немного привести очередь в порядок? Если в очереди имеется два или более одинаковых вопросов, записи с большими номерами можно пометить словом «duplicate» (повтор). Посмотрите, насколько актуально сообщение об ошибке, и если ошибка была исправлена, отметьте ее словом «fixed» (исправлено). Если вы знаете ответ на заданный вопрос – ответьте на него. Каждая минута, потраченная вами на подобную работу, – это время, которое разработчики смогут потратить на улучшение своего модуля, поэтому подобный вклад в проект высоко ценится разработчиками.

Помощь в поддержке пользователей

Если вы зашли так далеко, что сумели установить Drupal, примите наши поздравления! Теперь вы знаете достаточно, чтобы помочь кому-то другому. Посещайте форумы Drupal или канал *#drupal-support* на сайте *irc.freenode.net* и помогайте отвечать на вопросы других пользователей. В процессе этой деятельности вы узнаете много нового.

Работа над проблемами

Если вам довелось столкнуться с проблемой в модуле или вас посетила идея о добавлении новой «крутой» особенности, составьте подробный отчет об ошибке или детально опишите предлагаемую особенность в очереди вопросов модуля, следуя рекомендациям со страницы *http://drupal.org/node/317*. Не забудьте сначала воспользоваться ссылкой Advanced search (Расширенный поиск) в начале очереди во-

просов, чтобы убедиться, что подобного сообщения или предложения еще нет в очереди.

Документация

Не приходилось ли вам потратить лишние полчаса на решение проблемы из-за отсутствия документации или из-за отсутствия описания ошибки в существующей документации? Добавьте в документацию свои исправления, чтобы помочь сэкономить время тем, кто столкнется с такой же проблемой. Вы можете также присоединиться к группе документирования на странице *http://drupal.org/contribute/documentation/join*, чтобы принять участие в разработке документации для системы Drupal.

Пожертвования

У вас нет времени, чтобы лично принять участие в проекте, но у вас имеются свободные средства? Вы можете пожертвовать свои средства организации Drupal Association, которая предоставляет серверную инфраструктуру, организует конференции и производит сбор средств для проекта Drupal, на странице *http://association.drupal.org/donate*. Многие отдельные разработчики также с благодарностью примут пожертвования. Если вы пользуетесь чьим-то модулем, который помог вам сэкономить средства, имеет смысл передать некоторую их часть разработчику в качестве благодарности.

Почему стоит принять участие в проекте? Помимо «приятного чувства вовлеченности» есть еще ряд практических причин:

- Если вы являетесь участником проекта, то, как правило, вашим просьбам о поддержке, сообщениям об ошибках и предложениям будет уделяться большее внимание.
- Активное участие в проекте поможет вам устанавливать связи, которые могут привести вас к клиентам или работодателям.
- Участие в проекте позволит вам сэкономить месяцы на изучении Drupal – благодаря общению с людьми, с которыми иначе вы не встретились бы.
- Вы сможете помочь в определении направлений развития модулей и даже ядра Drupal, чтобы обеспечить более полное их соответствие вашим потребностям.
- Кроме того, это просто интересно! Вы будете встречаться с людьми со всего мира и учиться у самых светлых умов веб-дизайна.

С нетерпением ожидаем встречи с вами на сайте Drupal.org!

В заключение

Советы и рекомендации, приведенные в этом приложении, помогут вам найти модули, которые «должен иметь каждый», даже после того, как эта книга устареет. Оценивая такие параметры, как активность раз-

работчиков, количество участников проекта, полнота документации и качество оформления программного кода, вы сможете сделать правильный выбор при поиске модулей в будущем. А принимая участие в жизни сообщества, вы сможете встретить удивительных людей, которые развивают Drupal, и стать одним из них!

Ссылки

Ниже перечислены ресурсы, упоминавшиеся в этом приложении:

- Руководство по составлению отчетов об ошибках и запросов на разработку новых особенностей: *http://drupal.org/node/317*
- Примеры: *http://drupal.org/cases*
- Страница для вступления в проект: *http://drupal.org/contribute*
- Список сторонних модулей: *http://drupal.org/project/Modules*
- Репозиторий CVS: *http://cvs.drupal.org/viewvc.py/drupal/contributions/modules*
- Список разработчиков проекта Drupal: *http://drupal.org/project/developers/3060*
- Группа документирования: *http://drupal.org/contribute/documentation/join*
- Пожертвования: *http://association.drupal.org/donate*
- Очередь вопросов проекта ядра Drupal: *http://drupal.org/project/issues/drupal*
- Форумы Drupal.org: *http://drupal.org/forum*
- Форум «Before You Start» (Прежде чем начать): *http://drupal.org/forum/20*
- Форум Drupal, где представлены примеры сайтов: *http://drupal.org/forum/25*
- Пример сайта New York Observer: *http://drupal.org/node/141187*
- Планета Drupal: *http://drupal.org/planet*
- Пример сайта журнала Popular Science: *http://drupal.org/node/233090*
- Пример сайта Sony BMG MyPlay: *http://drupal.org/node/241344*
- Сторонние ресурсы: *http://drupal.org/node/289913*

C

Модули и темы оформления, использованные в этой книге

В этом приложении перечислены модули и темы оформления, использовавшиеся в примерах этой книги. Все они включены в комплект исходных текстов, которые можно загрузить с сайта книги, и перечислены здесь для справки или для случаев, когда вы хотели бы воспроизвести примеры, разрабатываемые в главах, без загрузки комплекта исходных текстов.

При написании этой книги использовалась версия Drupal 6.4.

Глава 1. Обзор системы Drupal

Нет использовавшихся модулей.

Глава 2. Запускаем Drupal

Модули:

- Administration menu 6.x-1.0 (*http://drupal.org/project/admin_menu*)
- Модуль FCKeditor 6.x-1.3-rc1 (*http://drupal.org/project/fckeditor*)
- Пакет FCkeditor 2.6.3 (*http://fckeditor.net*)
- IMCE 6.x-1.1 (*http://drupal.org/project/imce*)

Темы:

- Garland (входит в состав ядра)

Глава 3. Каталог вакансий

Модули:

- Advanced help 6.x-1.0 (*http://drupal.org/project/advanced_help*)
- Content Construction Kit (CCK) 6.x-2.0-rc10 (*http://drupal.org/project/cck*)
- FileField 6.x-3.0-alpha5 (*http://drupal.org/project/filefield*)
- Views 6.x-2.0-rc4 (*http://drupal.org/project/views*)

Темы:

- Wabi 6.x-1.1 (*http://drupal.org/project/wabi*)

Глава 4. Обзор продуктов

Модули:

- Advanced help 6.x-1.0 (*http://drupal.org/project/advanced_help*)
- Amazon 6.x-1.0-beta3 (*http://drupal.org/project/amazon*)
- Content Construction Kit (CCK) 6.x-2.0-rc10 (*http://drupal.org/project/cck*)
- CSS Injector 6.x-1.3 (*http://drupal.org/project/css_injector*)
- Fivestar 6.x-1.13 (*http://drupal.org/project/fivestar*)
- Search (входит в состав ядра)
- Views 6.x-2.0-rc4 (*http://drupal.org/project/views*)
- Voting API 6.x-2.0-beta6 (*http://drupal.org/project/votingapi*)

Темы:

- Nitobe 6.x-1.6 (*http://drupal.org/project/nitobe*)

Глава 5. Wiki

Модули:

- Advanced help 6.x-1.0 (*http://drupal.org/project/advanced_help*)
- Diff 6.x-2.0 (*http://drupal.org/project/diff*)
- Freelinking 6.x-1.6 (*http://drupal.org/project/freelinking*)
- Markdown filter 6.x-1.0 (*http://drupal.org/project/markdown*)
- Pathauto 6.x-1.1 (*http://drupal.org/project/pathauto*)
- Token 6.x-1.10 (*http://drupal.org/project/token*)
- Views 6.x-2.0-rc4 (*http://drupal.org/project/views*)

Темы:

- Barlow 6.x-1.0 (*http://drupal.org/project/barlow*)

Глава 6. Управление процессом публикации

Модули:

- Advanced help 6.x-1.0 (*http://drupal.org/project/advanced_help*)
- Trigger (входит в состав ядра)
- Views 6.x-2.0-rc4 (*http://drupal.org/project/views*)
- Views Bulk Operations 6.x-1.1 (*http://drupal.org/project/views_bulk_operations*)
- Workspace 6.x-1.3 (*http://drupal.org/project/workspace*)
- Workflow 6.x-1.1 (*http://drupal.org/project/workflow*)

Темы:

- Light Fantastic 6.x-1.0 (*http://drupal.org/project/lightfantastic*)

Глава 7. Фотогалерея

Модули:

- Advanced help 6.x-1.0 (*http://drupal.org/project/advanced_help*)
- Content Construction Kit (CCK) 6.x-2.0-rc10 (*http://drupal.org/project/cck*)
- Custom Pagers 6.x-1.10-beta1 (*http://drupal.org/project/custom_pagers*)
- FileField 6.x-3.0-alpha5 (*http://drupal.org/project/filefield*)
- ImageCache 6.x-1.0-alpha2 (*http://drupal.org/project/imagecache*)
- ImageField 6.x-3.0-alpha1 (*http://drupal.org/project/imagefield*)
- Token 6.x-1.10 (*http://drupal.org/project/token*)
- Views 6.x-2.0-rc4 (*http://drupal.org/project/views*)

Темы:

- Ubiquity 6.x-1.x-dev (от 31.05.2008) (*http://drupal.org/project/ubiquity*)

Глава 8. Многоязычные сайты

Модули:

- Book (входит в состав ядра)
- Content Translation (входит в состав ядра)
- Forum (входит в состав ядра)
- Internationalization 6.x-1.0-beta1 (*http://drupal.org/project/i18n*)
- Localization (входит в состав ядра)
- Localization client 6.x-1.3 (*http://drupal.org/project/l10n_client*)

Темы:

- Dreamy 6.x-1.3 (http://drupal.org/project/dreamy)

Глава 9. Управление событиями

Модули:

- Advanced help 6.x-1.0 (*http://drupal.org/project/advanced_help*)
- Content Construction Kit (CCK) 6.x-2.0-rc10 (*http://drupal.org/project/cck*)
- Calendar 6.x-2.0-rc3 (*http://drupal.org/project/calendar*)
- Date 6.x-2.0-rc3 (*http://drupal.org/project/date*)
- Flag 6.x-1.0-beta5 (*http://drupal.org/project/flag*)
- Views 6.x-2.0-rc4 (*http://drupal.org/project/views*)

Темы:

- Deco 6.x-1.1 (*http://drupal.org/project/deco*)

Глава 10. Интернет-магазин

Модули:

- Content Construction Kit (CCK) 6.x-2.0-rc10 (*http://drupal.org/project/cck*)
- FileField 6.x-3.0-alpha5 (*http://drupal.org/project/filefield*)
- ImageCache 6.x-1.0-alpha2 (*http://drupal.org/project/imagecache*)
- ImageField 6.x-3.0-alpha1 (*http://drupal.org/project/imagefield*)
- Token 6.x-1.10 (*http://drupal.org/project/token*)
- Ubercart 6.x-2.x-dev (от 3.11.2008) (*http://drupal.org/project/ubercart*)

Темы:

- Pixture 6.x-1.1 (*http://drupal.org/project/pixture*)

Глава 11. Оформление сайта

Модули:

- Devel 6.x-1.9 (*http://drupal.org/project/devel*)

Темы:

- Bluemarine (входит в состав ядра)

Алфавитный указатель

A

Administration Menu, модуль, 77
AdSense, модуль, 243
Akismet, веб-служба, 108
Amazon, модуль, 216
Apache, веб-сервер
    функция Clean URLs, 329
Attachment (прикрепленный файл), тип отображения, 166
Automatic Node Titles, модуль, 204
Auto Timezone, модуль, 397

B

Backlinks, представление, 277
block.tpl.php, файл шаблона, 502
Block (блок), тип отображения, 167
Blog, модуль, 45, 243
Body (содержимое), поле, 54
Book, модуль, 354
box.tpl.php, файл шаблона, 502

C

Calendar Popup, модуль, 432
Calendar, модуль, 421
Cart, модуль, 437
Catalog, модуль, 459
CCK (Content Construction Kit), пакет, 139
    параметры отображения полей, 241
    форматы, 327
Clean URLs, функция
    ImageCache, модуль, 329
    решение проблем, 329
comment.tpl.php, файл шаблона, 502
Community Tags, модуль, 349
Conditional Actions, модуль, 469
Content Permissions, модуль, 205
Content Translation, модуль, 354, 355
Countdown, модуль, 432
cron, утилита
    модуль Search, 235
CSS Injector, модуль, 239
Custom Links, модуль, 205
Custom Pagers, модуль, 346

D

Date, модуль, 401, 405
Devel, модуль, 389, 496
Diff, модуль, 270
Drupal
    благодарность разработчикам, 559
    загрузка, 526
    использование, 38
    каталоги, 528
    обзор, 29
    обновление, 535
    об обратной совместимости, 537
    помощь проекту, 560
    совместимость версий, 73
    установка, 524
    файлы, 528
    ядро, 52

E

e-Commerce, модуль, 436
Event, модуль, 401

F

FCKeditor, модуль, 127
Feed (рассылка RSS), тип отображения, 167
Firebug, расширение, 495
Fivestar, модуль, 211, 222
Flag Actions, модуль, 432
Flag, модуль, 305, 402, 426
Forum, модуль, 353
Freelinking, фильтр, 249

G

GD, библиотека, 330
GMap, модуль, 432
Google Analytics, служба
    для использования в интернет-магазинах, 444

I

i18n, модуль, 385
$Id, тег в файлах Drupal.org, 498
Image Assist, модуль, 121
ImageCache, модуль, 310, 321
ImageField, модуль, 310
IMCE, модуль, 122
.info, файлы
  оформление сайта, 486
install.php, сценарий, 528
Internationalization, модуль, 385

J

JavaScript
  оформление сайта, 491
JPEG, изображения
  миниатюры, 335

L

Language icons, модуль, 396
Liquid Wiki, модуль, 247
Locale, модуль, 355, 366, 371
Localization Client, модуль, 369, 372
Location, модуль, 432

M

Microsoft IIS, веб-сервер
  функция Clean URLs, 330
Mollom, веб-служба, 108
Mom and Pop, Inc., веб-сайт, постановка задачи, 51
Multilingual Blocks, модуль, 384
Multilingual Menu, модуль, 383
Multilingual Poll, модуль, 384
Multilingual Profile, модуль, 384
Multilingual taxonomy, модуль, 384

N

Nodequeue, модуль, 307
node.tpl.php, файл шаблона, 502
Notifications, модуль, 243

P

page.tpl.php, файл шаблона, 502
Page (страница), тип отображения, 167
Page, тип содержимого
  сравнение с типом Story, 250
Pathauto, модуль, 250, 261
Path, модуль, 75
PayPal, модуль, 480
PEAR Wiki Filter, модуль, 277
PHP
  доступ к изображениям, созданным с помощью модуля ImageCache, 328
  имена разрешений, 89
  модуль Date, 405
  проблемы безопасности, 61
  просмотр конфигурации, 330
.po, расширение имен файлов, 356
Poormanscron, модуль, 235
preprocess, функции, 513
Primary links, меню, 59
Profile, модуль, 384

R

Rules, модуль, 283

S

Search, модуль, 233
Secondary links, меню, 59
Secure Pages, модуль, 480
settings.php, файл, 386
SQL
  и модуль Views, 165
Stock, модуль, 480
Story, тип содержимого
  сравнение с типом Page, 250
Strings, модуль, 389
Sweet Tees, веб-сайт интернет-магазина, 435

T

Table of Contents, модуль, 277
Tagadelic, модуль, 349
Talk, модуль, 277
Taxonomy
  или CCK, 141
Taxonomy Redirect, модуль, 349
Taxonomy, модуль, 44, 99, 141
Theme Developer, модуль, 496
Tracker, модуль, 272
Translation Overview, модуль, 396
Translation Synchronization, модуль, 384
t(), функция, 356

U

Ubercart, пакет, 436
update.php, сценарий, 528, 540
Update Status, модуль, 537
Upload, модуль, 138

V

Views Bulk Operations, модуль, 301

Views, модуль, 164, 172
  организация поиска, 236
Voting API, модуль, 222

W

Web Developer Toolbar, панель инструментов, 496
wiki, 245
  адреса URL, 250
  модуль Tracker, 272
  настройка, 254
  настройка прав доступа, 252
  отображение информации об авторе, 252
  отслеживание изменений, 249
  отслеживание редакций, 267
  создание типов содержимого, 250
  ссылки, 249
  форматирование текста, 248
  форматы ввода, 253, 256
  функциональные возможности, 247
Wikitools, модуль, 247
Workflow Fields, модуль, 305
Workflow Graph, модуль, 307
Workflow, модуль, 291
Workspace, модуль, 299

А

административный раздел, настройка
  темы оформления, 116
адреса URL
  в wiki, 250
  изображений, 323
аргументы
  модуль Views, 170
атрибуты продуктов
  интернет-магазины, 450
базы данных
  обновление Drupal, 540
  требования, 525
безопасность
  PHP, 61
библиотеки
  GD, библиотека, 330
  модуль Date API, 405

Б

благодарность разработчикам, 559
блоги
  создание, 101
блоки, 46
  интернационализация, 384
  и темы оформления, 116
  настройка, 67
  определение, 59
блокирование пользователей, 85

В

ввода форматы, 122
веб-окружение
  установка на локальную систему, 526
виджеты
  CCK, 143
  ввода дат, 407
визуальные редакторы, 119
выгрузка
  файлов, 138
  фотографий, 310, 315

Г

главная страница, 44
  узлы, 56

Д

действия
  ImageCache, модуль, 323
  для процесса публикации, 285, 303
документация
  помощь, 561
дополнительные ссылки, меню, 59
доставка товаров, 469
доступ к фильтрам, 236

З

загрузка
  Drupal, 526
задачи
  Epic University, 137
  Migratory Patterns, веб-сайт, 353
  веб-сайт Super Duper Chefs (Супер-пупер повара), 208
  веб-сайт интернет-магазина Sweet Tees, 435
  веб-сайт компании Mom and Pop, Inc., 51
  веб-сайт фотогалереи семьи Робинсон, 309
  сайт Twin Cities Arts, 281
  сайт wiki студенческой ассоциации органов самоуправления в университете города Берхем, 247
  сайт общества книголюбов Aurora Book Club, 400
заказы
  интернет-магазин, 474

И

идентификаторы продуктов
поиск, 221
извещения
по электронной почте, 282
процесс публикации, 282
изменение размеров изображений, 324
изменения, отслеживание в wiki, 249
изображения, 335
качество, 335
обработка, 52, 120
импорт и экспорт информации из интернет-магазина, 444
инсталлятор
локализованный, 359
интернационализация
модули, 385
определение, 351
средства, 355
интернет-магазины, 434
Ubercart, пакет, 436
атрибуты продуктов, 450
веб-сайт Sweet Tees, 435
классы продуктов, 450
настройка, 448
покупательская корзина, 437, 467
продукты, 452
процесс оформления заказа, 464
информация об авторе
управление отображением, 252

К

кадрирование изображений, 324
каскадные таблицы стилей
оформление сайта, 489, 500
каталог вакансий, 136
CCK (Content Construction Kit), пакет, 139
Views, модуль, 164
каталоги
Drupal, 528
качество
изображений, 335
классификация
продуктов, 440
фотографий, 318
классы продуктов, 454
интернет-магазины, 450
комментарии, 45, 57
комплекты
товаров, 444
копирование тем оформления, 497
критерии сортировки
модуль Views, 172
курсивный шрифт, 168

Л

лексемы, 262
модуля Pathauto, 265
локализация
инсталлятора, 359
определение, 351

М

маркировка членами сообщества, 349
масштабирование изображений, 325
меню
Administration Menu, модуль, 77
интернационализация, 383
меню по умолчанию, 59
метаданные
для тем оформления, 486
миниатюры
создание, 310
мини-модули, 520
многоязычные сайты, 351
Book, модуль, 354
Forum, модуль, 353
Locale, модуль, 371
Localization Client, модуль, 369, 372
интернационализация, 355, 385
перевод содержимого, 360, 368, 376
переключение языка, 357
пользовательский интерфейс, 354, 370
содержимое, созданное пользователями, 354
модерирование содержимого
инструменты, 108
модули, 39
Administration Menu, 77
AdSense, 243
Amazon, 216
Automatic Node Titles, 204
Auto Timezone, 397
Blog, 45, 243
Book, 354
Calendar, 421
Calendar Popup, 432
Cart, 437
Catalog, 459
Community Tags, 349
Conditional Actions, 469
Content Permissions, 205
Content Translation, 354, 355

Countdown, 432
CSS Injector, 239
Custom Links, 205
Custom Pagers, 346
Date, 401, 405
Devel, 389, 496
Diff, 270
e-Commerce, 436
Event, 401
FCKeditor, 127
Fivestar, 211, 222
Flag, 305, 402, 426
Flag Actions, 432
Forum, 353
GMap, 432
i18n, 385
Image Assist, 121
ImageCache, 310, 321
ImageField, 310
IMCE, 122
Internationalization, 385
Language icons, 396
Liquid Wiki, 247
Locale, 355, 366, 371
Localization Client, 369, 372
Location, 432
Multilingual Blocks, 384
Multilingual Menu, 383
Multilingual Poll, 384
Multilingual Profile, 384
Multilingual taxonomy, 384
Nodequeue, 307
Notifications, 243
Path, 75
Pathauto, 250, 261
PayPal, 480
PEAR Wiki Filter, 277
Poormanscron, 235
Rules, 283
Search, 233
Secure Pages, 480
Stock, 480
Strings, 389
Table of Contents, 277
Tagadelic, 349
Talk, 277
Taxonomy, 44, 99, 141
Taxonomy Redirect, 349
Theme Developer, 496
Tracker, 272
Translation Overview, 396
Translation Synchronization, 384
Update Status, 537
Upload, 138
Views, 164, 172
Views Bulk Operations, 301
Voting API, 222
Wikitools, 247
Workflow, 291
Workflow Fields, 305
Workflow Graph, 307
Workspace, 299
обновление, 543
определение, 71
отключение, 75
поиск, 72
поиск и выбор, 546
список, 563
страница администрирования, 71
удаление, 75
установка, 72
модули API, 219
мультисайтинг, 530

Н

наборы правил
ImageCache, модуль, 323, 327, 332
навигационное меню, 59
создание, 66
навигационные цепочки
функции тем оформления, 514, 517
шаблоны, 503
навигация
определение, 59
налоги
в интернет-магазинах, 444, 469
настройка
блоков, 67
действий для процесса публикации, 287
модуля Amazon, 217
модуля Catalog, 459
модуля CSS Injector, 241
модуля FCKeditor, 127
модуля Flag, 427
модуля Pathauto, 264
неповторимого внешнего вида сайта, 117
отображения поля Photo, 334
пакета Ubercart, 448
пользователей, 85
прав доступа, 81, 252
представлений, 169
представления для процесса публикации, 304
разрешений, 88
разрешений формата, 257

тем оформления, 112
темы оформления административного раздела, 116
фильтров в wiki, 254
настройки видимости блоков на языке PHP, 521
нумерация версий
Drupal, 535

О

обзоры продуктов, 207
модуль Amazon, 216
модуль CSS Injector, 239
модуль Search, 233
модуль Voting API, 222
первый шаг, 211
список продуктов, 228
облака тегов, 349
области
и темы оформления, 116
оформление сайта, 487, 511
области для блоков
отображение, 488
обновление
Drupal, 535
системы безопасности, 539
обратная совместимость, 537
основные ссылки, меню, 59
особенности
Drupal, 32
отключение
модулей, 75
отношения
модуль Views, 172
отображение
изображений, 310
информации о продукте, 209
модуль Views, 166
областей для блоков, 488
полей, 154
отображение полей
параметры, 241
отслеживание
изменений в wiki, 249
процесса публикации, 282
редакций в wiki, 267
отчеты
об ошибках, 560
о состоянии дел в интернет-магазине, 442, 445
очереди вопросов, 559
оформление заказов
в интернет-магазине, 464
оформление сайта, 482
собственные темы, 496
файлы шаблонов, 491, 502
оценка
продуктов, 210
оценки посетителей
включение, 225
добавление, 224
очереди вопросов, 552
помощь в приведении в порядок, 560

П

панели инструментов
Web Developer Toolbar, 496
модуля FCKeditor, 132
редактирование содержимого, 119
перевод
Locale, модуль, 355, 366
пользовательского интерфейса, 354, 356, 370
синхронизация таксономии и узлов, 384
содержимого, 354, 360, 368, 376
установка, 361
переключение языка на многоязычных сайтах, 357, 366
переменные сайта, 386
интернационализация, 386
переменные шаблонов, 513
оформление сайта, 513
переопределение, 516
переопределение
переменных шаблонов, 516
строк, 387
тем оформления, 513
файлов шаблонов, 508
функций тем оформления, 515
Планета Drupal, веб-сайт, 549
платежи
обработка, 441
платформы управления содержимым, 30
пожертвования
проекту Drupal, 561
поиск
модулей, 72
тем оформления, 111
покупательская корзина
интернет-магазины, 437, 467
пользователи, 40
блокирование, 85
настройка прав доступа, 81
настройки, 85

поддержка, 560
управление доступом, 80
пользовательский интерфейс
перевод, 354, 356, 370
поля, 142
CCK, 142
модуль Views, 169
модуля Amazon, 218
отображение, 154, 166
поле Rating, 224
права доступа
определение, 80
представления
Calendar, 422
примеры
популярных веб-сайтов на базе Drupal, 548
приращение времени, 411
продукты
интернет-магазины, 452
классификация, 440
произвольная маркировка
для фотографий, 319
просмотр
изображений, обработанных набором правил, вручную, 328
редакций в wiki, 269
профили пользователей
определение, 84
процесс публикации, 280
действия и триггеры, 285
изменение настроек по умолчанию, 283
модули, 291

Р

радиокнопки, 143
разработчики Drupal, 556
разрешения
для редакций, 269
добавление, 214
настройка, 88
при создании роли, 82
управление доступом, 81
форматы ввода, 257
раскрывающийся список, 144
редактирование
содержимого, 52, 119
редакции
отслеживание, 267
режим профилактики сайта, 539
резервное копирование
при обновлении Drupal, 541
репозиторий программного кода Drupal, 555
решение проблем
ImageCache, модуль, 329
функция Clean URLs, 329
роли, 41
для пользователей, выполняющих покупки, 442
создание, 86
управление доступом, 81

С

свойства
оформление сайта, 488
узлов, добавление с помощью модулей, 43
синонимы
Pathauto, 263
синхронизация
таксономии и узлов, 384
системы управления содержимым, 33
слежение
за численностью участников, 402
за уровнем запасов в интернет-магазине, 445
совместимость
модулей, 547
содержимое, 53
определение, 53
перевод, 354, 360, 368, 376
поддержки, 45
редактирование, 52, 119
создание, 63
создание
блогов, 101
блока для отображения, 200
действий для процесса публикации, 303
наборов правил для ImageCache, 332
навигационного меню, 66
областей, 511
представлений, 189
процесса публикации, 294
рабочих пространств, 300
ролей, 86
собственных тем оформления, 496
содержимого, 63
списков, 138
страницы отображения, 191
типа содержимого Event (Событие), 403
типов содержимого, 158, 145, 212, 250, 403

узлов, 42
формата ввода wiki, 256
формы контактов, 96
сообщения об обновлениях, 534
спам
определение, 108
списки
материалов в wiki, 273
создание, 138
список продуктов в интернет-магазине
поиск, 236
создание, 228
список ссылок для главы
каталог вакансий, , 206, 279, 397
ссылки
в wiki, 249
получение справки, 46
установка и обновление Drupal, 545
страница администрирования модулей, 71
страницы
главная страница, 44
обзорные страницы, 283
проектов, 550
страница администрирования модулей, 71
страница поиска, 211
сценарии
обновления Drupal, 540
установки Drupal, 528

Т

таксономия
интернационализация, 384, 393
определение, 99
текст
перевод пользовательского интерфейса, 354
форматирование в среде wiki, 248
текстовое поле, 143
с функцией автодополнения, 144
текстовые редакторы
оформление сайта, 495
темы оформления
настройка, 112
определение, 111
поиск, 111
установка, 111
тестовые пользователи, 83
тестовый заказ
интернет-магазины, 471
типы полей для хранения дат, 406
типы представлений
Calendar, 422
типы содержимого
Event (Событие), 403
Job Application (заявка), 158
Job (вакансия), 146
в CCK, 145
для wiki, 250
для обзоров продуктов, 212
для фотографий, 315
интернационализация, 389
оформление, 507
точность
представления дат, 412
требования
Drupal, 525
триггеры
назначение, 290
процесс публикации, 285

У

удаление
данных, 75
информации об авторе при отображении, 252
модулей, 75
узлы, 41
продукты в пакете Ubercart, 450
сравнение с блоками, 62
узлы страниц
создание, 54
управление доступом
интернет-магазины, 477
управление событиями, 399
модуль Calendar, 421
модуль Flag, 426
права доступа, 404
представление Attendee (Участники), 429
представление для отображения предстоящих событий, 417
слежение за количеством участников, 402
тип представления Calendar, 422
типы содержимого, 403
управление содержимым
инструменты, 53
установка
Drupal, 524
веб-окружение, 525
модулей, 72
модуля FCKeditor, 127

перевода, 361
тем оформления, 111
устранение проблем
ресурсы, 47
участие в проекте Drupal, 560
участники
слежение за численностью, 402

Ф

файлы
.css, 489
Drupal, 528
.info, 486
install.php, 528
.js, 491
settings.php, 386
update.php, 528, 540
загружаемые, 441
файлы переводов, 356
шаблонов, 491, 502
фильтры
Freelinking, 249
Freelinking, фильтр, 254
PEAR Wiki Filter, 277
в wiki, 254
и форматы ввода, 122
модуль Views, 170
настройка, 254
организация доступа, 236
порядок следования, 256
флажки, 143
форма контактов
создание, 96
форматирование
текста, 248
форматы, 155
CCK, 144
форматы ввода, 122
wiki, 253, 256
дат, 411
и фильтры, 122
разрешения, 257
формы ввода
CCK, 138
форумы
интернационализация, 393
форумы Drupal.org
поиск и выбор модулей, 548
фотогалереи, 308
Custom Pagers, 346
ImageCache, модуль, 321
ImageField, модуль, 310
выгрузка фотографий, 310
создание миниатюр, 310
фрагменты
блоков на языке PHP, 520
страниц, 521
функции
preprocess, 513
t(), 356
тем оформления, 514, 517

Ч

часовые пояса, 406, 413, 534
чистые ссылки
настройка сервера, 534

Ш

шаблонов файлы
оформление сайта, 491, 502
шрифт
курсивный, 168

Э

экспорт и импорт информации из интернет-магазина, 444
электронная почта
извещение, 282
этапы
процесса публикации, 282, 292, 294

Я

ядро
Drupal, 52
языки с письмом справа налево, 355

По договору между издательством «Символ-Плюс» и Интернет-магазином «Books.Ru – Книги России» единственный легальный способ получения данного файла с книгой ISBN 978-5-93286-174-5, название «Drupal: создание и управление сайтом» – покупка в Интернет-магазине «Books.Ru – Книги России». Если Вы получили данный файл каким-либо другим образом, Вы нарушили международное законодательство и законодательство Российской Федерации об охране авторского права. Вам необходимо удалить данный файл, атакже сообщить издательству «Символ-Плюс» (piracy@symbol.ru), где именно Вы получили данный файл.