

УДК 378

Особенности создания электронных дидактических материалов с помощью системы управления содержимым Drupal

С.Д. Коткин

Новокузнецкий филиал (институт) Кемеровского государственного университета,
пр. Пионерский 13, Новокузнецк, Россия

sergeykotkin@gmail.com

Статья поступила 10.10.2016, принята 18.12.2016

В статье рассматривается история возникновения систем управления содержимым веб-сайта в процессе общей эволюции информационных систем. В данном случае под эволюцией понимается добавление уровней абстрагирования содержания от механизмов реализации, вследствие чего появляется возможность декомпозиции задач по созданию и эксплуатации информационных систем. Описаны ключевые особенности системы управления содержимым веб-сайта Drupal, обосновано применение метода проектов для создания электронных дидактических материалов с применением этой системы.

Ключевые слова: системы управления содержимым; метод проектов; Drupal.

Features of using content management system Drupal for creation of electronic didactic materials

S.D. Kotkin

Novokuznetsk Branch (Institute) of Kemerovo State University; 13, Pionersky Ave., Novokuznetsk, Russia

sergeykotkin@gmail.com

Received 10.10.2016, accepted 18.12.2016

The article deals with the history of the creation of content management systems in the process of the evolution of information systems through the transition from a set of static documents to generated ones based on user requests and data stored in information systems, as well as adding abstraction levels of content from implementation mechanisms, which results in the possibility of decomposition of tasks on creation and operation of these systems. The key features of the content management system of Drupal web site are described. The application of the project method for creation of electronic didactic materials when using this system is justified.

Keywords: content management systems; project method; Drupal.

Одной из задач, решаемых в ходе подготовки современных студентов по направлению «технология», является обучение созданию электронных дидактических материалов. В большинстве случаев средой их распространения служит интернет, а сами электронные дидактические материалы создаются в виде веб-сайта, который является примером информационной системы.

Еще в 60–70-х гг. XX в. подавляющее большинство информационных систем представляли собой базы данных, выполняющиеся локально на мейнфреймах, однако уже с 80-х гг. в связи с массовым внедрением персональных компьютеров, а также ростом доступности сетей передачи данных, структура информационных систем стала меняться. Первым таким изменением стало появление распределенных систем, изна-

начально основанных на двухуровневой архитектуре «клиент – сервер», где каждый компьютер может выполнять роль клиента или сервера. Компьютер-клиент решал такие задачи организации взаимодействия вычислительной системы с пользователем, как организация ввода данных, первичная обработка пользовательского ввода и создание представления результатов обработки данных. Сама же обработка выполнялась на серверах — компьютерах, которые получали запросы от клиентов, обрабатывали на основании этих запросов данные из баз данных и возвращали результаты обработки клиенту. Такая архитектура позволила создавать гетерогенные сети обработки данных, в которых могли взаимодействовать информационные системы уровня организации и персональные информационные системы. В дальнейшем развитие пошло по пути увеличения количества отдельных уровней системы, в частности, серверы разделились на серверы баз данных и серверы приложений, добавлялись возможности создания реплик баз данных для функционирования географически распределенных систем, появилась система балансировки нагрузки на вычислительные кластеры. Следующий шаг в своем развитии информационные системы совершили в связи с появлением интернета: общедоступная глобальная сеть дала толчок унификации способов взаимодействия отдельных компонентов систем путем использования прикладных протоколов интернета. Корпоративные сети, в которых отдельные компоненты взаимодействовали с помощью открытых протоколов интернета, стали называться intranet-сетями. Сейчас развитие информационных систем идет по пути использования «облачных» технологий, в которых обработка и хранение данных осуществляются однородными распределенными вычислительными кластерами, обеспечивающими путем резервирования надежность системы и осуществляющими автоматическую балансировку нагрузки для обеспечения максимальной производительности.

Одним из способов облегчить начало изучения современных информационных систем на примере веб-сайтов является использование метода проектов. Коллективные проекты по разработке управляемых

данными веб-сайтов позволяют как понять место изучаемого материала в общей картине предметной области через представление о своей роли в результате реализации проекта (что, несомненно, важно в связи с высокой сложностью современных информационных систем и даже их отдельных составляющих), так и мотивировать студента путем вовлечения в продуктивную коллективную деятельность [1].

Первые веб-сайты представляли собой набор файлов, к которым веб-сервер обращался при обработке запроса клиента. Основным типом передаваемых с сервера файлов были HTML-документы — текстовые файлы, в которые добавлены теги — специальные последовательности символов, с помощью которых браузеру указывается, как обрабатывать помеченный этими тегами текст. Однако по мере расширения масштабов применения интернета появлялись и новые задачи, в частности потребность двустороннего взаимодействия с пользователями, с тем чтобы они могли управлять серверами не только путем выбора определенных гиперссылок, но и вводя данные, такие как текст и числа, которые сервер смог бы обработать, а затем отправить на компьютер пользователя ответ, зависящий от результатов обработки. Например, поисковые машины позволяют пользователю ввести запрос и выдают страницы со ссылками, содержание которых определяется, в том числе, и текстом этого запроса. Вначале подобное двунаправленное взаимодействие пользователя с веб-сервером было, как правило, сосредоточено на небольшом количестве страниц сайта, что было вызвано сложностью организации подобного взаимодействия, обусловленной новизной решаемых при этом задач и, как следствие, недостаточной их технологизированностью: отсутствием общепринятых подходов к решению, дефицитом программистов, способных такие задачи решать, а также неразвитостью подходящих инструментальных средств. Однако со временем острота этих проблем стала снижаться, что привело к взрывному росту численности сайтов с возможностью отправки на них данных. Но это также привело к тому, что подходы к разработке сайтов в целом стали меняться.

Изначально HTML был разработан для представления научных и технических текстов внутри сетей научно-исследовательских и учебных организаций. Для этого набор тегов, используемых для структурирования текста, был весьма небольшим, что упрощало его изучение. Были ограничены и возможности по форматированию элементов текста — способ их отображения определялся браузером, программой для просмотра веб-страниц. Однако со временем HTML начал применяться для создания сайтов различных организаций и частных лиц, предназначенных для широкого круга посетителей. Для этих сайтов было характерно изменение способа навигации. Если большинство HTML-документов представляло собой преимущественно обычные тексты, иногда иллюстрированные, где отдельные их фрагменты являлись гиперссылками, позволяющими осуществлять навигацию по сайту, то в сайтах для массового посетителя навигация приобрела первостепенное значение: в связи с тем, что общий объем документов в сети стал быстро увеличиваться, удобство поиска информации на сайте определяло то, насколько долго посетитель на нем останется. Поэтому, помимо гиперссылок в тексте, стали создаваться специальные элементы — меню, присутствующие на каждой странице и позволяющие переходить к основным разделам сайта. Возросла также важность оформления: посещаемость сайта стала определяться не только его содержанием, но и внешней привлекательностью.

Эти два изменения сделали поддержку сайта весьма трудоемкой задачей. Во-первых, меню для навигации повторялось на каждой странице сайта, соответственно, любое изменение его структуры отражалось на содержании всех страниц. Во-вторых, изменение оформления сайта в пору становления стандартов HTML, сопровождающегося выработкой новых подходов к оформлению сайтов, быстро меняющихся технических возможностей компьютеров и глобальных сетей передачи данных, а также так называемой «войны браузеров» (создания расширений HTML, увеличивающих возможности языка разметки, но поддерживаемых только одним из браузеров) требовалось производить весьма часто. Для упрощения

задач поддержки было предпринято несколько мер. У большинства веб-серверов появилась возможность компоновать один HTML-документ из нескольких файлов. В этом случае такие элементы, как меню, могли размещаться в отдельном файле и включаться в содержание всех страниц. Также произошли изменения со стандартом HTML — все большее число браузеров стало поддерживать каскадные таблицы стилей — CSS, которые позволяли отделить описание внешнего вида элементов HTML-документа от содержимого этих элементов и вынести часть дизайна в отдельные файлы, содержавшие только описание оформления текста, но не сам текст документов. Однако все эти решения не давали возможность полностью разделить структуру, оформление и средства навигации сайта, оставляя необходимым, пусть и в несколько меньшей степени, изменять значительную часть содержания сайта при изменении одного из этих компонентов.

Новый подход к созданию сайтов заключался в том, что вместо статических страниц сайт стал представлять собой набор программ, зачастую написанных на интерпретируемых языках, таких как Perl, PHP или Visual Basic. Программы запускались в ответ на запрос пользователя и генерировали веб-страницу, которую и возвращал в качестве ответа web-сервер. При этом результаты, как правило, получались путем соединения данных, хранящихся в отдельных файлах или базах данных, и шаблонов представления страниц, определявших, как эти данные отображать. Сами же программные компоненты отвечали за взаимодействие с пользователем. Помимо довольно четкого разделения компонентов сайта, такой подход позволял, во-первых, распараллелить процесс создания сайтов, давая возможность разработчикам решать одну из самостоятельных задач, а во-вторых, реализовать интерактивное взаимодействие с посетителем не на отдельных страницах, а на всем сайте. Например, можно было дать посетителю возможность написать комментарий к статье или выполнить поиск информации на сайте, не обращаясь к поисковикам. Такого рода системы управления

содержимым веб-сайта получили название CMS (Content Management System).

Изначально CMS разрабатывались либо с нуля, либо с использованием в качестве заготовок наработок из предыдущих проектов, что требовало как знания HTML и сетевых протоколов, так и навыков программирования и делало невозможным создание сайтов на основе CMS неспециалистами в области информационных технологий. Со временем отдельные CMS эволюционировали в завершённые программные продукты, которые могли использовать не только их разработчики. Поначалу тиражируемыми стали специализированные системы для организации форумов, чатов, доступа к почте, позднее появились системы общего назначения.

Одной из таких CMS является Drupal [2; 3]. Эта система начала разрабатываться в 2000 г. в качестве «доски объявлений» и к 2001 г. стала проектом с открытым исходным кодом. К настоящему времени Drupal превратился в CMS общего назначения и представляет собой модульную систему, то есть, его функциональные возможности реализуются посредством модулей, которые взаимодействуют с веб-сервером, базой данных и друг с другом через ядро системы. Такая модульная архитектура позволила сделать Drupal адаптируемым к требованиям конкретного сайта. С учетом же того, что помимо базовых модулей, входящих в состав системы наравне с ядром, имеется обширная библиотека готовых модулей различного назначения, настройка системы для реализации заданных требований во многих случаях возможна без дополнительного программирования. Можно выделить следующие особенности Drupal: возможность множественной иерархической классификации

всех публикуемых материалов; система контроля прав доступа, развитие которой в имеющихся модулях позволяет формализовать процессы публикации материалов на сайте; возможность создания новых типов материалов посредством описания структуры представляющих эти материалы данных; возможность публикации материалов параллельно на нескольких языках и прочее, что отличает современные веб-сайты.

Использование Drupal как платформы для реализации электронного дидактического средства позволяет организовать занятия с адаптацией их содержания и структуры к особенностям как учебных групп в целом, так и отдельных студентов, начиная с реализации структуры сайта путем использования ограниченного количества базовых возможностей и итеративно включая дополнительные функциональные модули по мере осознания их необходимости, а также роста уровня владения представлениями об интерактивных веб-системах в общем и о Drupal в частности. Это дает возможность изначально формировать у студентов целостное представление о реализуемых на веб-платформах дидактических средствах.

Литература

1. Чупин Д.Ю., Ростовцев А.Н. Реализация продуктивного обучения в процессе подготовки будущих учителей технологии и предпринимательства // Сибирский педагогический журнал. 2007. № 7. С. 121-130.
2. Описание CMS Drupal [Электронный ресурс] // CMS Magazine: [сайт]. URL: [http:// drupal.cmsmagazine.ru/](http://drupal.cmsmagazine.ru/) (дата обращения: 04.11.2016).
3. Drupal (Друпал) [Электронный ресурс] // TAdviser: [сайт]. URL: <http://www.tadviser.ru/index> (дата обращения: 07.11.2016).