

УДК 004.415.2

UDC 004.415.2

КОРПОРАТИВНАЯ ИНФОРМАЦИОННО-СПРАВОЧНАЯ СИСТЕМА ВЫСШЕГО УЧЕБНОГО ЗАВЕДЕНИЯ

CORPORATION INFORMATION-INQUIRY SYSTEM OF EDUCATIONAL ESTABLISHMENT

Бульбанюк Артур Игоревич

Bulbanuk Arthur Igorevich

Нестерова Нонна Семёновна
к.т.н., доцент

Nesterova Nonna Semenovna
Cand.Tech.Sci., associate professor

Академия маркетинга и социально-информационных технологий (ИМСИТ), Краснодар, Россия

Academy of Marketing and Social-Information Technologies (IMSIT), Krasnodar, Russia

В статье дана концепция построения информационно-справочной системы ВУЗа и приведены некоторые результаты ее реализации

The article tells about the conception of creating information-inquiry system of the institute and the results of its work

Ключевые слова: WEB-ТЕХНОЛОГИИ, ДОМЕН, ОБЪЕКТНО-ОРИЕНТИРОВАННЫЙ ПОДХОД, АВТОМАТИЗИРОВАННАЯ СИСТЕМА, МОДЕЛЬ ДАННЫХ, МОДУЛЬ СИСТЕМЫ

Keywords: WEB-TECHNOLOGIES, DOMAIN, OBJECT-ORIENTED APPROACH, AUTOMATIC SYSTEM, DATA MODEL, SYSTEM MODULE

В связи с необходимостью повышения качества управления учебным процессом в академии маркетинга и социально-информационных технологий (г. Краснодар) была поставлена задача создания и внедрения корпоративной информационно-справочной системы, предоставляющей всю учебно-методическую, распорядительную и прочую документацию студентам, преподавателям и менеджерам отдельных подразделений ВУЗа.

К ней предъявлены следующие требования:

- каждое зарегистрированное подразделение имеет возможность создания своей «стартовой» страницы и тематических разделов, характеризующих его деятельность и отражающих последние изменения,
- задачи администрирования распределены между подразделениями и группами пользователей,
- инструментарий управления располагает средствами для свободного перемещения структурных элементов между различными тематическими разделами, блоками и подразделениями без повреждения внутренних или внешних

связей. Сюда можно отнести перемещение отдельных параграфов и файлов, передачу дисциплин и информации между различными кафедрами, создание и перестановку дочерних подразделений в случае учреждения, расформирования или иной реорганизационной деятельности внутри ВУЗа.

Дополнительные возможности:

- поддержка удалённого доступа,
- регистрация новых типов подразделений.

Особо следует отметить необходимость смешанного режима работы, в котором система воспринимает и структурирует материалы всех дочерних подразделений как один большой набор данных. Это позволяет найти информацию, не зная наименования связанного подразделения – при нахождении искомого элемента программа самостоятельно осуществляет переключение на нужный дочерний ресурс.

Система разрабатывается силами преподавателей кафедры компьютерных систем управления и обработки информации и студентов учебно-исследовательской лаборатории «Проектирование программного обеспечения», созданной при ней.

Для максимального соответствия приведённым ранее требованиям, проект ориентирован на использование web-технологий. В задачи разработчиков входят также изучение, обобщение и синтез принципов построения серверных приложений с целью выявления наиболее эффективных решений, пригодных к многократному использованию.

В качестве сервера использован Internet Information Services 7.5. Для реализации серверной логики применен объектно-ориентированный подход и практичный С-подобный скриптовый язык программирования PHP, что обусловлено его открытой архитектурой и возможностью достаточно лёгкого освоения. В роли сервера баз данных выступил

продукт Microsoft SQL Server 2008 Express, а для «общения» с ним использован язык Transact-SQL и драйвер SQL Server Driver for PHP от Microsoft, предоставляющий разработчикам быстрый и компактный API-интерфейс взаимодействия.

Первоначально сформулирована идея создания информационного ресурса одной кафедры, однако в процессе проработки его структуры и выделения значимых частей стало понятно, что схожая абстрактная модель применима к большинству управляемых сайтов. Поэтому возник вопрос, как заставить несколько сайтов работать на одном наборе программного кода, чтобы минимизировать трудоёмкость процесса развёртывания и сопровождения дополнительных ресурсов.

Рассмотренные варианты реализации предусматривают ручное или автоматическое выделение системой домена нижнего уровня для каждого подразделения учебного заведения (например, адрес кафедры компьютерных систем управления и обработки информации `ksu.imsit.local` будет являться частью домена `imsit.local`).

Подобный подход сильно ограничивает возможности будущей системы, поскольку возникает вопрос о корректном перемещении материалов между кафедрами, факультетами, институтами и прочими подразделениями. Любой перенос страницы между доменами приведёт к повреждению существующих ссылок, так как в этом случае (при неизменной конфигурации директорий) привязка формируется на основе двух компонент – домена подразделения и идентификатора некоторой страницы.

Для небольших сайтов допустима переадресация. Однако в системе, которая представляет всё учебное заведение в целом, должны без особых сложностей производиться манипуляции объёмными блоками данных с множеством внутренних связей. Так, если представить дисциплину и её электронные материалы как подверженную частым изменениям структуру

с неограниченным числом иерархически упорядоченных элементов, то при перемещении такой дисциплины или отдельных ветвей иерархии между подразделениями учебного заведения любая переадресация окажется ненадёжной и потеря отдельных ссылок – лишь вопрос времени.

Разработанная нами автоматизированная система под условным названием «Ками» лишена упомянутых недостатков.

В ней все физически существующие страницы (их относительно немного) представлены небольшими наборами контроллеров, которые в зависимости от полученного на входе «сигнала-привязки» обращаются к базе данных за тем или иным наполнением. В роли источника привязки выступает ядро системы (рисунок 1), в составе которого присутствует специальное «наборное устройство», которое анализирует строку входящего запроса и подготавливает сведения, необходимые для дальнейшей работы сценария. В случае, если наборным устройством получен идентификатор некоторой страницы, то в процессе подтверждения существования будут выявлены достоверные идентификаторы комплекса, дисциплины и подразделения, которым принадлежит запрашиваемая страница.

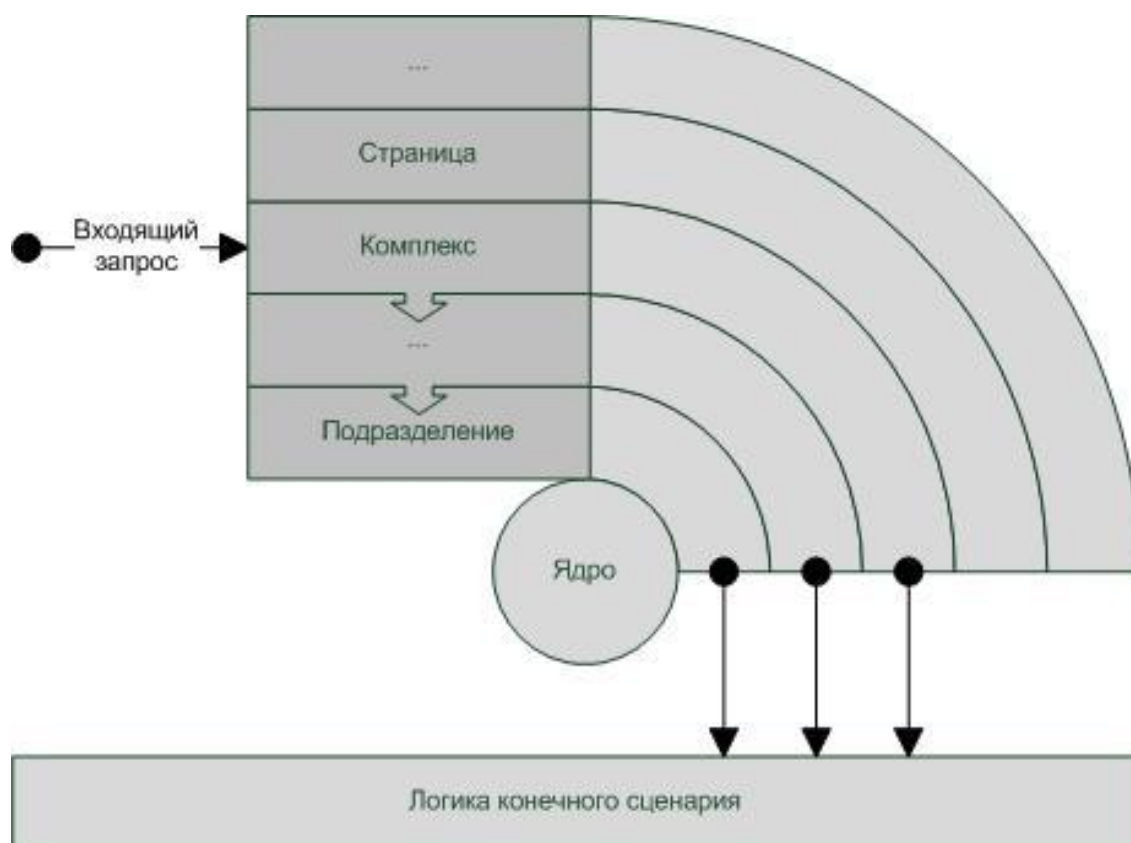


Рисунок 1 – Принцип работы ядра

Для защиты от фальсификации идентификаторов (злоумышленник указывает одновременно страницу и подразделение, где он имеет права администрирования) система выполняет приоритетную инициализацию: сначала производится сканирование адресной строки всеми звеньями ядра (рекурсивный полиморфный спуск к конструктору базового класса); затем устанавливается соединение с базой данных и все звенья проверяют свои идентификаторы. Если высший слой ядра (страница) определил, что он не содержит информации о каком-либо низшем слое (комплексе, дисциплине, подразделении), то соответствующий слой удаляет свои идентификаторы и на все обращения к атрибутам отвечает значением null. В противном случае производится замещение потенциально ложных идентификаторов.

После того, как наборное устройство завершает инициализацию, оставшаяся часть ядра идентифицирует и авторизует пользователя,

загружая из базы информацию о его полномочиях в активном подразделении.

В зависимости от потребностей сценария в него может быть включен тот или иной уровень наборного устройства.

В штатном режиме генерируемые компонентами страницы ссылки содержат идентификатор, указывающий на необходимый и достаточный уровень цепи, поэтому число задействованных узлов стремится к минимуму.

Особенности применяемой модели данных (рисунок 2) делают любую перестановку внутренней, изменения не отражаются на ссылках.

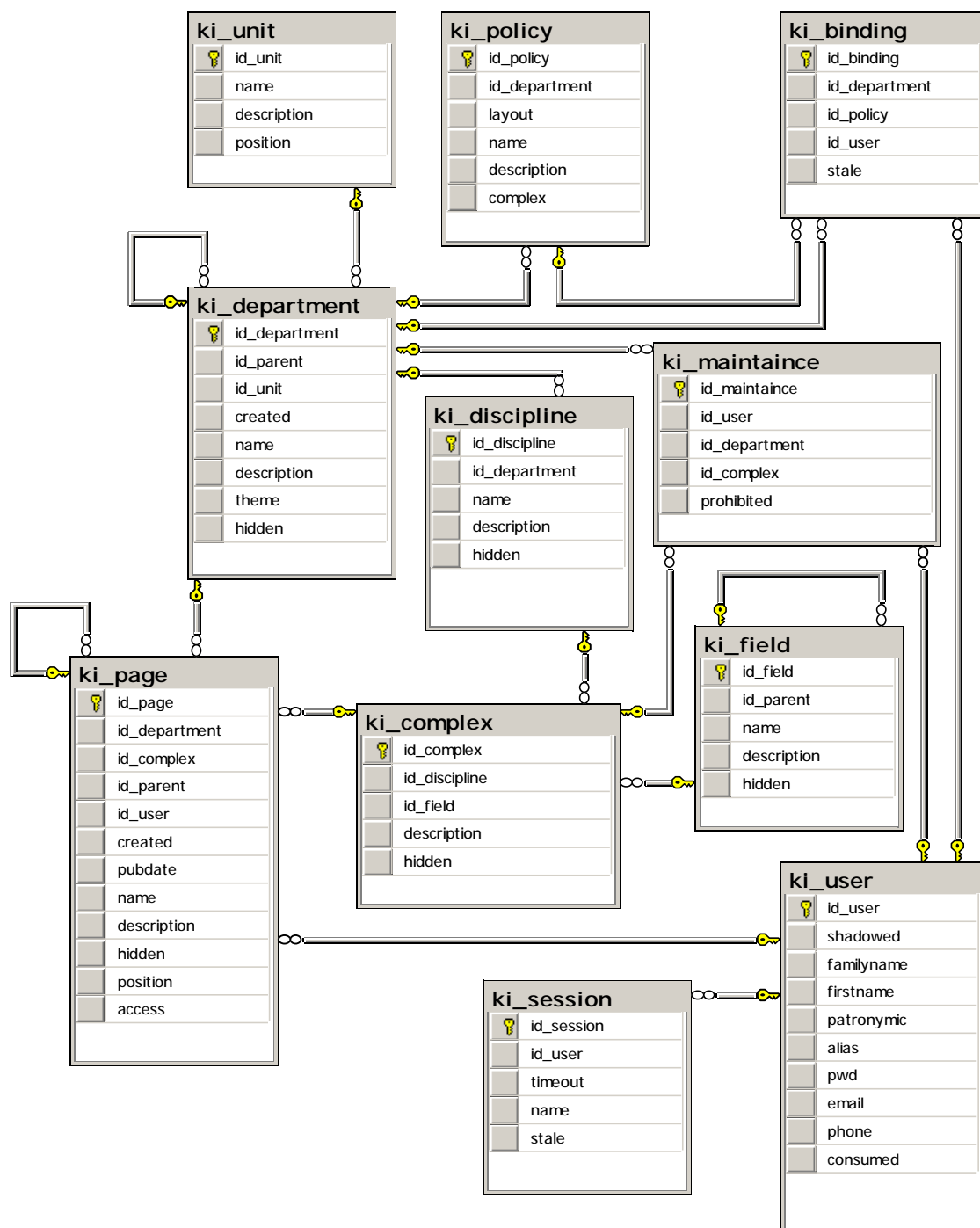


Рисунок 2 – Общая модель кластера данных

Например, если со страницей или иным структурным элементом в момент перемещения велась какая-либо работа, то при повторном обращении к серверу может поменяться заголовок подразделений или карта пройденного пути, но ссылка на эту страницу будет по-прежнему действительной.

Соответствующая как известным, так и определённым в процессе разработки внутренним стандартам и соглашениям, Ками поддается лёгкой реконструкции с целью расширения функциональных возможностей. Прозрачное применение профессиональных техник оптимизации делает процесс передачи данных максимально быстрым и стабильным, а следование стандартам консорциума W3C в области гипертекстовой разметки (X)HTML5 обеспечивает возможность мгновенного переключения тем оформления для различных подразделений. Стилиевые правила могут быть разработаны профессиональными дизайнерами, остающимися в неведении относительно конструкций программного кода.

В последних версиях ядра поддерживается параллельная работа с несколькими подразделениями на разных вкладках браузера (ранние реализации для хранения информации об активном подразделении использовали сеансовый кэш, что накладывало некоторые ограничения).

Автоматизированный архив - основной модуль системы (рисунок 3). Информация в нём делится на порции, оформленные в виде иерархически связанных страниц.



Рисунок 3 – Интерфейс управления архивом

Каждая страница допускает создание неограниченного числа текстовых блоков (рисунок 4) и позволяет прикреплять файлы в любых форматах.

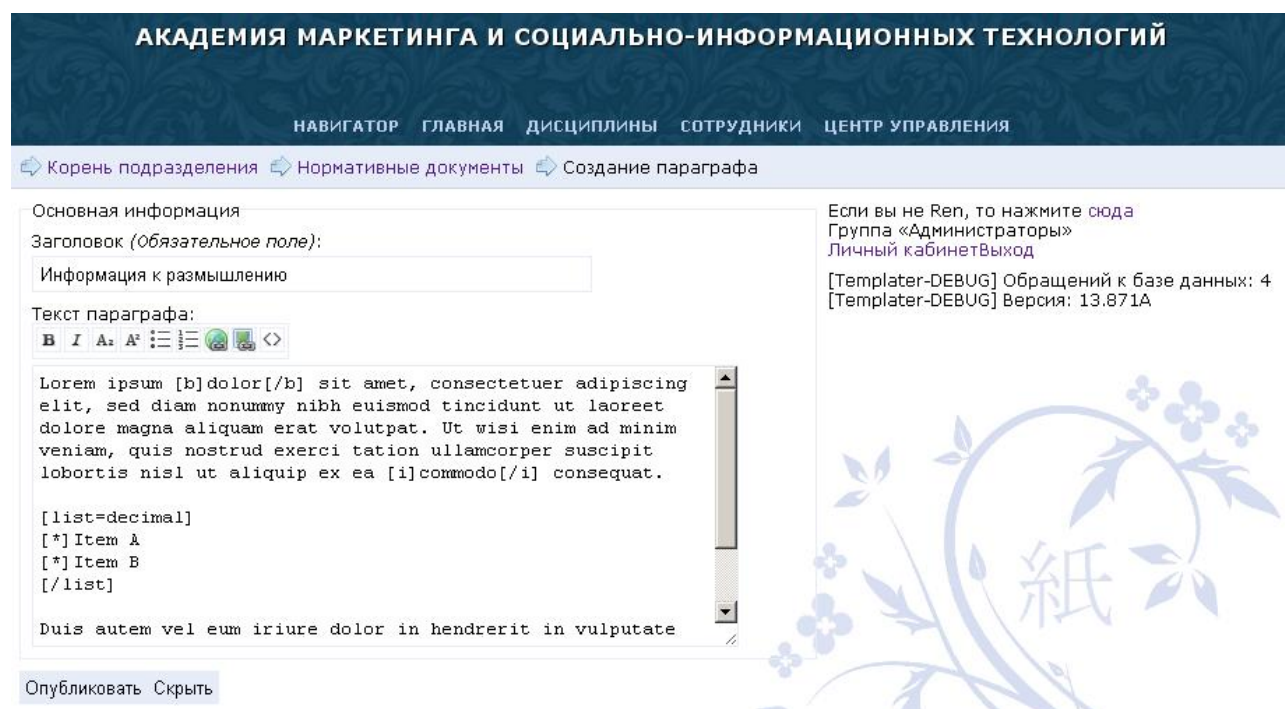


Рисунок 4 – Создание параграфа

Все загруженные файлы сохраняются в базе данных, а доступ к их содержимому производится при помощи технологии FILESTREAM, которая объединяет компонент Database Engine с файловой системой NTFS, размещая данные больших двоичных объектов в файловой системе. Используется системный кэш NT, что позволяет снизить возможное влияние данных FILESTREAM на производительность компонента Database Engine [2].

Компоненты архива:

- *поисковая система* – позволяет найти информацию по заголовкам, текстовому наполнению страниц и файлов,
 - *новостная лента* – выводит список недавно созданных страниц.
- Благодаря интеграции ранее независимой новостной подсистемы появилась возможность создания иерархически

упорядоченных новостей и тематических разделов, новостные сообщения публикуются естественным образом,

- *список сотрудников подразделения* – отражает информацию о связанных с подразделением пользователях системы, позволяет переходить на полуавтоматические персональные страницы, содержащие сведения о преподавателях, календарь с планом проведения занятий, список закреплённых дисциплин и прочую информацию. Часть заполняется самим сотрудником, а часть генерируется на основе прав доступа к элементам панели управления.

В ходе исследования выделены основные компоненты информационно-справочной системы, определены сущности и связи модели данных. Дальнейшее развитие предполагает введение дополнительных модулей-расширений, каждый из которых отвечает за те или иные функции некоторого типа подразделений учебного заведения.

СПИСОК ЛИТЕРАТУРЫ

1. Кузнецов, М.В. РНР. Практика создания web-сайтов [Текст] / М.В. Кузнецов, И.В. Симдянов. – СПб.: БХВ-Петербург, 2008 – 1264 с.
2. Общие сведения о FILESTREAM [Электронный ресурс]: Microsoft Software Development Library. – Режим доступа: <http://msdn.microsoft.com/ru-ru/library/bb933993.aspx>
3. Котеров, Д. В. РНР 5 [Текст] / Д. В. Котеров, А. Ф. Костарев. – СПб.: БХВ-Петербург, 2008. — 1104с.
4. Фленов, М. Е. Transact-SQL [Текст] / М. Е. Фленов – СПб.: БХВ-Петербург, 2006. — 576с.
5. Могилев, А.В. Информатика [Текст]: Учебное пособие для студ. педвузов / А.В. Могилев [и др.]; под ред. Е. К. Хеннера. – М.: Академия, 2006. – 326с.
6. Советов, Б.Я. Моделирование систем [Текст] / Б.Я. Советов, С.А. Яковлев – М.: Высш.шк., 2007. – 343с.