УДК 519.6

СИСТЕМА “АРЕОЛА” — ПРОГРАММНАЯ ОБОЛОЧКА ДЛЯ СОЗДАНИЯ ЭЛЕКТРОННЫХ ЭНЦИКЛОПЕДИЙ

П. А. Брызгалов[1]

В статье описывается система, представляющая собой программную оболочку для создания Интернет-энциклопедий по различным областям знаний.

В настоящее время ведутся активные исследования в различных областях электронных справочных изданий. Это обусловлено стремлением использовать практически неисчерпаемые возможности вычислительных машин по обработке различного рода информации. Важнейшим фактором, способствующим распространению электронных изданий, является Интернет. Публикация издания в Интернете существенно отличается от публикации книжного издания своей доступностью, динамичностью и возможностью публиковать не только тексты и графику, но и видео, звук и сделать электронное издание интерактивным. Рассматриваемая система ориентирована на реализацию возможности электронных изданий отражать внутреннюю логическую структуру предметной области.

Система “Ареола” представляет собой электронную оболочку, позволяющую создавать справочные электронные издания по различным предметным областям и публиковать их в Интернете.

Многие предметные области (как естественнонаучные, так и гуманитарные) можно представить в виде совокупности отдельных статей. Все статьи можно разбить на группы по нескольким принципам. Во-первых, по типу: определения, факты, аксиомы, теоремы, свойства, комментарии и т.п. Во-вторых, если предметная область достаточно широка, статьи в ней можно разделить по разделам и главам, как это бывает в книгах. И в-третьих, статьи можно разделить иерархически по сложности, т.е., условно говоря, статьи, которые необходимо знать на “тройку”, которые надо знать на “четверку”, на “пятерку” и на “пять с плюсом”. Предметные области, допускающие подобное разбиение на статьи и их классификацию, подходят для построения электронных энциклопедий при помощи данной системы.

Как уже упоминалось, одной из главных задач системы является отражение внутренней логической структуры предметной области. Что под этим подразумевается? Во многих предметных областях между статьями существуют те или иные отношения: например, одни статьи используют факты, содержащиеся в других, или с одной статьей предпочтительно ознакомиться после другой, или что-то иное. Все такие отношения будем называть логическими. Если в предметной области можно установить логические отношения, то такая предметная область может быть представлена направленным графом, в котором вершины — это статьи, а из одной вершины в другую ведет дуга тогда и только тогда, когда вторая статья опирается в логическом отношении на первую. Будем называть статью, из которой выходит дуга, “предшественником”, а в которую входит — “следствием”. Будем предполагать, что этот граф является ациклическим. Начальными вершинами должны стать аксиомы и основные определения, которые опираются на то, что для рассматриваемой предметной области считается уже известным и выходит за рамки энциклопедии. Такой граф для каждой предметной области не однозначен, и каждый специалист может построить свой граф.

Система “Ареола” позволяет заложить в энциклопедию информацию о подобных логических отношениях между статьями и имеет ряд функций, позволяющих использовать логические отношения для более эффективного изучения материала. Эти функции включают визуализацию графа логических отношений, поиск предшественников и следствий и нахождение статей, которые лежат на всех возможных путях между несколькими заданными “опорными” статьями. Подробнее обо всех функциях системы будет сказано ниже.

Работа пользователя с системой заключается в поиске и выборе нужных статей и отображении информации об этих статьях и их логических взаимосвязях. Существуют следующие возможности поиска статей:

— поиск по структурному указателю,

[1] Научно-исследовательский вычислительный центр, Московский государственный университет им. М.В. Ломоносова, 119992, Москва; e-mail: pyotr777@guru.ru

— поиск по предметному указателю,
— поиск по ключевым словам,
— поиск статей по номерам,
— переход от статьи к статье по ссылкам,
— поиск логических предшественников и следствий,
— ограничение поиска по статьям с уровнем сложности не выше заданного,
— поиск по графу логических отношений статей.

Заметим, что в традиционных книгах из этих восьми возможностей доступны только четыре: первая, вторая, четвертая и пятая. Но даже и этими возможностями в книге не так удобно пользоваться, поскольку необходимо ее перелистывать в поисках нужного номера страницы или статьи. Различие в удобстве использования хорошо понимает любой, кто пользовался обычными словарями и их электронными версиями. Поиск нужного слова в электронном словаре требует гораздо меньше сил и времени.

Чтобы выбрать статью, пользователь отмечает ее галочкой в соответствующем поле. Номера выбранных статей образуют так называемые "выборки". Существует четыре разные выборки, которые соответствуют четырем основным способам выбора статей: структурному указателю, предметному указателю, поиску по ключевым словам и поиску по номерам. Выборки во время работы хранятся в оперативной памяти компьютера пользователя. Пользователь может применять к этим выборкам операции сложения, вычитания и пересечения множеств. Полученная в результате "объединенная" выборка всегда загружается в структурный указатель.

Пользователь может сохранить выборки, как отдельные, так и "объединенную", у себя на компьютере, чтобы продолжить работу с ними позже. Кроме упомянутых выше существуют еще две функции: первая — так называемая "функция пополнения". Пользователь отмечает в структурном указателе "опорные" статьи: статьи, которые он уже знает или понимает — начальные статьи, и статьи, которые ему надо понять — конечные статьи. Применяя функцию пополнения к опорным статьям, он получает все статьи, которые лежат на всех возможных путях между опорными статьями в графе логических отношений. Добавленные статьи автоматически отмечаются в структурном указателе. Таким образом, пользователь отмечает все статьи, которые ему необходимо изучить, чтобы понять конечные. Другая функция позволяет найти всех предшественников всех статей, отмеченных в одном из четырех основных режимов поиска и выбора статей.

Отображение информации из выбранных статей и информации об их логических взаимосвязях может осуществляться четырьмя основными способами. К первой группе — функциям просмотра статей — относятся функции просмотра текстов статей для чтения с экрана и для распечатки на принтере. Ко второй группе относятся функции просмотра графа логических отношений выбранных статей и вывод списка предшественников или следствий одной или нескольких статей. Отображение осуществляется в новых окнах браузера, что позволяет пользоваться различными функциями системы по выбору статей и отображению информации одновременно.

Таким образом, каждый способ просмотра и выбора статей (структурный и предметный указатели, поиск по ключевым словам и поиск по номерам, просмотр текстов статей на экране и вывод их на экран для печати и, наконец, просмотр графа логических отношений) отображается в отдельном окне браузера. Исключение составляют предметный указатель и поиск по словам и по номерам, которые отображаются в одном и том же окне (но не одновременно).

Тексты статей и список предшественников или следствий снабжены полями для пометки, состояние которых соответствует выборке структурного указателя. Иными словами, поля возле тех статей, которые входят в выборку, отмечены галочками, а возле статей, которые не входят, — пустые. Отмечая невыбранные статьи, пользователь добавляет их к выборке структурного указателя, а убирая галочки возле них, исключает их из выборки. Таким образом осуществляется синхронизация выбранных статей между окнами браузера с текстами статей со списком предшественников или следствий и со структурным указателем.

Рассмотрим техническую сторону реализации системы "Ареола". Пользователь работает с системой через ее веб-интерфейс при помощи интернет-брузера. Управляет всей системой программа-сервер. Когда пользователь активизирует кнопки и ссылки интерфейса системы, генерируются запросы на определенные HTML-файлы, хранящиеся на сервере. HTML-файлы могут содержать вставки на языке сценариев PHP, а запросы пользователя могут содержать параметры. Эти запросы вместе с параметрами передаются программе-серверу, которая сначала выполняет команды языка PHP, если такие присутствуют в файле. Вставки на PHP позволяют получать информацию из базы данных, представлять ее в требуемом формате, перенаправлять запрос на другой файл и выполнять другие действия, необходимые для удо-

влетворения запроса пользователя. Выполнив действия, запрограммированные PHP-сценарием, сервер возвращает пользователю HTML-файл, который содержит запрошенную пользователем информацию в требуемом виде. Этот файл отображается в интернет-браузере пользователя. Кроме требуемой информации HTML-файлы содержат ссылки и кнопки, позволяющие генерировать новые запросы серверу.

Таким образом, работа пользователя состоит из циклов: при активизации ссылок веб-страницы генерируются запросы, которые отправляются серверу, обрабатываются им и возвращаются пользователю. В окне браузера пользователя отображается новая веб-страница, позволяющая посылать новые запросы серверу.

Информация в энциклопедии организована в иерархическую структуру: разделы, главы, параграфы и статьи. Группы статей образуют параграфы, группы параграфов образуют главы, а группы глав образуют разделы. Разделы, главы и параграфы будем называть общим словом "подраздел". Вся информация о предметной области, введенная в систему, хранится в базе данных. В главной таблице базы хранится информация обо всех подразделах и статьях энциклопедии. Эта таблица содержит следующие поля: номер раздела, номер главы, номер параграфа, номер статьи в главе, номер-строку, название, текст, дополнение, тип, уровень сложности, индекс, номера предшественников и количества предшественников и следствий для каждого уровня сложности. Рассмотрим все поля подробнее.

Номер раздела — номер того раздела, в который входит статья или подраздел. Устанавливается в ненулевое значение во всех строках таблицы. Для разделов это — номер самого раздела.

Номер главы — номер той главы, в которую входит статья или подраздел. Для разделов он не определен.

Номер параграфа — номер того параграфа, в который входит статья. Для параграфов — номер самого параграфа внутри главы. Для разделов и глав не определен.

Номер статьи — номер статьи внутри главы. Хотя статьи группируются по параграфам, номер параграфа не входит в номер статьи, т.е. статьи имеют сквозную нумерацию внутри главы.

Номер-строка — составное строчное представление номера подраздела или статьи. Для разделов он состоит из одного числа — номера самого раздела. Для глав — это номер раздела, точка, номер главы в разделе. Для параграфов — это три числа, разделенные точками: номер раздела, номер главы и номер параграфа внутри главы. Для статей — это тоже три числа, только первые два (номер раздела и номер главы) разделены точкой, а перед последним числом (номером статьи в главе) ставится тире.

Такая нумерация статей позволяет сравнительно просто перегруппировывать статьи по параграфам, что при формировании энциклопедии приходится делать довольно часто.

Название — название подраздела или статьи.

Текст — текст статьи. Для подразделов поле не определено.

Дополнение — дополнительная текстовая информация. Это могут быть комментарии к статьям, доказательства, примеры и так далее. Для подразделов это поле не определено.

Тип — тип статьи: определение, дополнение, утверждение и т.п. Для подразделов это поле также не определено.

Уровень сложности — уровень сложности статьи. Для подразделов это поле не определено. Хотя при работе с графами логических отношений есть возможность отображать уровень сложности подразделов, но там он вычисляется во время работы и определяется как высший уровень сложности входящих в подраздел статей, но только тех, что присутствуют в выборке, по которой строится граф. Таким образом, отображаемый в графе уровень сложности подразделов будет зависеть от набора статей, по которым строится граф.

Индекс — число, определяемое на основе иерархического положения статей и подразделов, которое служит для упорядочивания записей из таблицы.

Номера предшественников — список номеров-строк логических предшественников через запятую. Это поле используется в основном для удобства создателей системы, а при работе системы информация о предшественниках берется из другой таблицы, состоящей из двух полей: номер-строка предшественника и номер-строка следствия. Такую таблицу проще использовать для поиска предшественников и следствий и работа с ней происходит быстрее.

Количество предшественников и следствий — это восемь полей, которые строятся по количеству уровней сложности (четыре) для предшественников и для следствий. Не все поля бывают заполнены. Если уровень сложности статьи выше минимального, то поля, соответствующие предшественникам и следствиям с меньшим уровнем сложности, не заполняются, так как в этом случае сама статья никогда не будет использоваться.

На основе информации о подразделах из главной таблицы формируется структурный указатель

(оглавление). Кроме этого, в базе данных в отдельной таблице хранится предметный указатель — список терминов и соответствующих им номеров статей.

Вся текстовая информация представлена в формате HTML — стандарте представления информации в сети Интернет. В текст можно вставлять рисунки. Это могут быть пояснительные иллюстрации, фотографии, специальные символы, математические формулы и т.д. Более того, можно даже использовать мультимедийные вставки и многое другое: все виды информации, доступные для представления в Интернете. Все это позволяет делать язык разметки HTML. Непосредственно в текст вставляются ссылки на объекты, а сами объекты хранятся в виде файлов в указанных в ссылках директориях на сервере.

Для работы с системой пользователю нужен Интернет-браузер — программа, которая умеет отображать информацию в формате HTML и позволяет использовать язык JavaScript. Таким браузером может быть Internet Explorer, Mozilla или Netscape четвертой или седьмой версий.

Одним из ключевых понятий для пользователя при работе с системой является "выборка" — набор статей. Номера статей, составляющих выборки, хранятся на компьютере пользователя и работа с ними происходит без их передачи на компьютер-сервер. Это сделано для того, чтобы, во-первых, уменьшить нагрузку на сеть и избавить пользователя от долгих ожиданий реакции системы на его действия и, во-вторых, чтобы уменьшить нагрузку на сервер. Почему важно разгрузить сервер? Сервер обслуживает всех пользователей, и поэтому лишняя нагрузка на него будет увеличиваться пропорционально количеству работающих в данный момент. Превышение же нагрузки над возможностями компьютера-сервера будет приводить к длительным паузам в его работе, к отказам на запросы пользователей или даже к остановке его работы. Вот почему работа с выборками перенесена на компьютеры пользователей, а сделано это при помощи языка JavaScript, встроенного практически во все современные браузеры. JavaScript является единственным средством, позволяющим работать со всей информацией, отображаемой в браузере, без необходимости обращаться за помощью к серверу. При помощи этого средства осуществляется, в частности, добавление статей в выборки, хранение выборок на компьютере пользователя во время работы, переключение режимов работы и синхронизация полей выбора статей между разными окнами браузера (то есть между разными режимами работы).

Рассмотрим подробнее, как происходит работа в рамках языка JavaScript. Все переменные и функции скрипта привязываются к определенному окну браузера, но можно получить доступ из функций одного окна к функциям и переменным другого. Так, все четыре выборки — выборка структурного указателя, выборка предметного указателя, выборка поиска по ключевым словам и выборка ввода номеров — хранятся в окне структурного указателя. Вот почему во время работы это окно должно постоянно оставаться открытым. Есть еще одна причина для этого: обмен данными может осуществляться не между любыми окнами, а возможен только в том случае, если второе окно было открыто из первого при помощи функций языка JavaScript. В "Ареоле" окно структурного указателя является главным, порождающим все остальные окна: окно предметного указателя, поиска и ввода номеров, окна с текстами статей, окно с предшественниками или следствиями одной статьи и окно с предшественниками одной из четырех выборок, окно с Java-аплетом для визуализации графа логических отношений и окно с текстами статей для распечатки на принтере. Синхронизация с выборкой структурного указателя в окнах текстов статей, предшественников и следствий осуществляется следующим образом: при открытии нового окна происходит обращение к выборке структурного указателя. Как упоминалось выше, все выборки хранятся в окне структурного указателя. Доступ к переменным этого окна получить возможно, потому что именно это окно открывает все новые. Получив данные о выборке структурного указателя, скрипт нового окна проверяет все поля для отметки статей, которые присутствуют в этом новом окне, и отмечает те из них, которые соответствуют статьям, входящим в выборку структурного указателя.

Другой важной функцией языка JavaScript, о которой еще не говорилось, является получение данных из Java-аплета для отображения графа логических отношений. Между аплетом и функциями языка JavaScript того же окна возможен обмен данными. Таким образом, в частности, из Java-аплета передаются номера статей, выбранные внутри аплета, тексты которых мы хотим посмотреть, и окно с текстами этих статей открывается при помощи функций языка JavaScript.

К сожалению, у языка JavaScript есть недостатки. Во-первых, это медленная работа. Например, операция "очистить выборку" структурного указателя на компьютере с процессором Celeron 700MHz при больших выборках может занимать до 50 секунд. Во-вторых, это большие различия между реализациями языка в разных браузерах. На настоящий момент не существует общедоступных и распространенных альтернатив языку JavaScript. Остается надеяться, что в новых версиях браузеров реализация языка JavaScript будет приближаться к стандарту и будет быстрее работать.

Отображение графа логических отношений реализовано при помощи Java-аплета. Вершины графа

представляются в виде цветных прямоугольников, а логические отношения между статьями — в виде стрелок. Изначально граф изображается в виде ярусно-параллельной формы, причем если граф достаточно большой, на экране видны только несколько начальных уровней формы. Вершины раскрашиваются либо в цвет уровня сложности соответствующих статей (по умолчанию), либо по цветам уровней ярусно-параллельной формы. Можно изменять количество видимых на экране уровней, сдвигать уровни, изменять масштаб графа (изменяя размеры вершин-прямоугольников и максимальное количество видимых на экране уровней). Можно строить граф логических отношений между параграфами и главами. При этом из одной вершины графа параграфов или глав в другую идет дуга тогда и только тогда, когда существует хотя бы одна дуга, ведущая из статьи одного подраздела в статью другого в графе логических отношений статей. Можно выделять вершины и оставлять на экране видимыми только те дуги, которые ведут в или из выбранных вершин, и только те вершины, которые связаны с выделенными. Как упоминалось выше, из аплета можно вызывать тексты выбранных статей.

Остановимся кратко на подготовке материалов для наполнения системы. На первом этапе подготавливается один или несколько текстовых файлов в определенном формате с информацией о номерах статей и подразделов, номерах параграфов (для статей), названиях статей и подразделов, текстах, дополнениях, уровнях сложности, типах статей и о статьях-предшественниках. Текстовая информация может содержать разметку HTML.

После того, как текстовый файл или файлы подготовлены, запускается специальная программа. С ее помощью вся информация из текстовых файлов заносится в базу данных. Далее, используя эту же программу, мы генерируем вспомогательную информацию, необходимую для работы системы, которая также хранится в базе данных, а кроме того — файлы структурного указателя (по одному для каждого уровня сложности). Эти файлы надо положить в директорию, где находятся остальные файлы системы “Ареола”.

Следующий этап — проверка корректности логических отношений и уровней сложности статей. Во-первых, граф логических отношений не должен содержать циклов. Во-вторых, следуя по путям в графе логических отношений, уровень сложности статей не должен уменьшаться. Другими словами, нельзя, чтобы сложная статья имела простые следствия.

Для проверки наличия описанных выше ошибок и для внесения изменений в базу данных без привлечения посторонних программных средств служит другая специальная программа. Эта программа выводит информацию об ошибках и номера соответствующих статей, позволяет вызвать из базы данных содержимое любого атрибута любой статьи, отредактировать его и записать обратно в базу данных. Обе упомянутые программы написаны на языке Java.

Система “Ареола” начала создаваться в 1999 году. Для ее разработки использовались технологии баз данных, язык серверных скриптов, язык разметки HTML, язык JavaScript динамического изменения информации из веб-страницы на компьютере пользователя и язык Java, использовавшийся для рисования графа логических отношений, а также программа для конвертации текста из формата TEX в формат HTML.

Примером использования системы “Ареола” является электронная энциклопедия по линейной алгебре “Линеал”. Проблемы, связанные с разбиением предметной области на статьи и установкой логических связей между ними на примере линейной алгебры, описаны в [1]. Энциклопедия “Линеал” доступна для работы через Интернет (http://lineal.guru.ru).

При подготовке этой энциклопедии мы столкнулись с проблемой отображения математических формул в браузере. В настоящее время их поддержка доступна в экспериментальном режиме только в новых версиях браузеров, а несколько лет назад, когда система только начала создаваться, такой поддержки не было вообще. Поэтому пришлось все формулы и специальные символы отображать в виде картинок. В создании таких картинок очень пригодилась упомянутая программа конвертации из TEX в HTML. Она автоматически преобразовывает формулы, не поддающиеся представлению в текстовом виде, в картинки и вставляет их в текст, делая нужную разметку HTML.

Работа частично поддержана грантом РФФИ 03–07–90427.

СПИСОК ЛИТЕРАТУРЫ

1. *Воеводин В.В., Воеводин Вл.В., Брызгалов П.А.* О некоторых проблемах компьютеризации знаний. Тезисы докладов Всероссийской научной конференции “Научный сервис в сети Интернет”, Новороссийск. М.: Изд-во МГУ, 2000.

Поступила в редакцию
11.02.2005