

ИНТЕРАКТИВНАЯ БИБЛИОТЕКА АЛГОРИТМОВ И ПРОГРАММ АККОРД ДЛЯ ЗАДАЧ МАТЕМАТИЧЕСКОГО МОДЕЛИРОВАНИЯ*

С.П. ГОЛОЛОВОВА, В.И. КОЗЛОВ

Институт вычислительной математики и математической геофизики СО РАН

Новосибирск, Россия

e-mail: sweta@lapasrv.sscs.ru

Актуальность данной работы вызвана накоплением интеллектуального багажа сотрудников ИВМиМГ в виде алгоритмов и программ, развитием объектно-ориентированного программирования, сетевых технологий и желанием с помощью вторых сделать общедоступным первое.

Электронная библиотека АККОРД включает в себя алгоритмы по основным разделам численного анализа, цифровой обработке данных, методам оптимизации. Информация об алгоритмах размещается в специальной базе данных, содержащей статьи с описанием алгоритмов, библиографические данные, авторские рекомендации по применению в различных приложениях, исходные модули и исполняемые компоненты в формате DCOM[1]. Особенность данной библиотеки заключается в том, что доступ к ее ресурсам осуществляется из глобальной сети через интерфейсные гипертекстовые (HTML) страницы. Эти страницы служат бланками для формирования пользовательских запросов, так же, для отображения результата выполнения запроса. Так как одной из основных технологий, используемых при разработке интерфейсных страниц, является технология Active Server Pages (ASP) [2], имеется возможность запуска исполняемых модулей, подготовленных в технологии ActiveX, с различными исходными данными, задаваемыми пользователем. Результаты расчетов могут быть взяты с сервера посредством обычного FTP-клиента, включаемого во все интернет-браузеры. В дальнейшем предусмотрена разработка ActiveX Controls для графического представления входных данных и результатов расчетов на интерфейсных страницах

1. Введение

За последние 15 лет произошли качественные изменения как в создании программного математического (в полном смысле этого слова) обеспечения, так и в Интернет-технологиях. Компьютерная математика эволюционизировала от пакетов прикладных программ к математическим системам программирования (МСП) с более широким кругом решаемых задач, с развитым интерфейсом, символьными преобразованиями и графическим представлением результатов[3]. В МСП включаются, помимо ядра системы и общего интерфейса, библиотеки модулей и набор этих библиотек зависит от потребностей пользователя. В связи с бурным развитием объектно-ориентированного программирования и применения его при создании библиотек стало возможным создание и использование вычислительных модулей на разных платформах. В Интернет-технологиях [4] за эти же годы произошло прямо таки революционное изменение значения понятия «документ» (в том числе статья, журнал, книга). На заре развития Интернета это понятие означало или печатный документ или электронный документ в ASCII- кодировке. В начале девяностых, появились составные документы, что приблизило значение и внешний вид электронных документов к их печатным аналогам. Но по настоящему уникальным и незаменимым электронный документ стал тогда, когда он стал мультимедийным. Текст, изображение, звук и видео позволяют всесторонне отразить свойства объекта, которому посвящен документ. Особенно это важно в образовании, так как позволяет включить различные механизмы восприятия информации.

Но даже если под документом мы понимаем объемную книгу или многотомное издание, это не достаточно для полноценного изучения предмета. Как правило, изучая какой-либо объект или технологию, вы стараетесь получить информацию о предмете с различных точек зрения. Для реализации таких возможностей в конце восьмидесятых была предложена технология гипертекстовых документов. Эта технология позволяет создавать связанные между собой наборы документов со средствами навигации от одного документа к другому.

В начале девяностых гипертекстовая технология была распространена на глобальную компьютерную сеть. Эта технология получила название World Wide Web. С появлением специальных поисковых серверов пользователь получил доступ к документам по всему миру. В середине девяностых гипертекстовая технология была объединена с мультимедийной технологией. Так появился стандарт HTML и интерпретаторы файлов HTML. Но только после дополнения этой технологии интерфейсами с базами данных, разработчики

*Работа выполняется при поддержке грантов РФФИ по информационным технологиям №99-07-90422 и №01-07-90367

электронных документов смогли предоставить пользователю возможность самому формировать документ, компоновать наборы документов, искать нужные программные компоненты и запускать их на выполнение.

Как видно из выше изложенного, современные средства разработки документов позволяют создавать мощные, но в тоже время удобные и эффективные, информационные и обучающие ресурсы. К сожалению, в основном эти средства применяются в интернет-вариантах СМИ, рекламе, электронной торговле и биржевых операциях. Конечно, в мировой сети есть много электронных библиотек, посвященных различным областям знаний. Но это в основном набор электронных вариантов печатных документов, не всегда используемых законным образом.

В данной статье мы предлагаем свое видение организации и функционирования электронных интерактивных библиотек, которые могли бы использоваться в высшей школе и учреждениях, специализирующихся на разработке высокотехнологической продукции, на примере проекта «Интерактивная библиотека алгоритмов и программ АККОРД для задач математического моделирования». Функциональная структура и архитектурные особенности библиотеки, за исключением структуры таблиц базы данных, изображаются при помощи диаграмм языка UML [5] версии 1.3 (Unified Modeling Language).

2. Структура библиотеки

Современные сетевые приложения типа клиент-сервер настолько далеко ушли в своём развитии от своих предшественников, что даже получили название многоуровневых приложений. В многоуровневой модели информационных систем [2] обработка данных распределена между клиентом и сервером. Основные задачи, которые решает такая система, это: а) представление данных; б) бизнес-логика; в) службы хранения данных.

На уровне представления данных пользователь может не только взаимодействовать с приложением, вводить и выводить данные, но и управлять данными. В веб-технологии задачи уровня представления данных выполняет веб-браузер.

Службы хранения данных обеспечиваются структурированными хранилищами информации (серверами баз данных).

Между этими двумя уровнями находится область для разработки распределенных приложений. Уровень бизнес-логики задает правила управления обработкой приложений.

Трехуровневая структура изолирует каждый сегмент функциональных возможностей. Представление не зависит от правил обработки и бизнес-логики, которая, в свою очередь, является отдельной от данных. Эта модель требует намного больше затрат на анализ и проектирование, но значительно уменьшает расходы на техническую поддержку и сопровождение и, в конечном счете, увеличивает функциональную гибкость.

В соответствие с этой структурой разработана и библиотека Аккорд.

Компоненты наполнения электронной библиотеки подобного рода, можно разделить на четыре группы: а) утилиты поддержки базы данных, б) интерфейс с пользователем, в) поиск в базе данных, г) расчетные компоненты.

Утилиты поддержки базы данных – это обычные исполняемые модули той операционной системы, которая выбрана в качестве операционной системы для рабочей станции администратора библиотеки. В нашем случае – это MS Windows. Вряд ли стоит специально останавливаться на функциональных особенностях этих утилит. Они выполняют стандартный набор операций, необходимых для обновления базы данных, поэтому ограничимся перечислением только прецедентов, специфических для нашей библиотеки.

На рисунке 1 представлены варианты утилит поддержки базы данных, в качестве которой предполагается использовать сервер БД ORACLE, на данный момент используется СУБД MS ACCESS.

Интерфейс с пользователем происходит посредством HTML страниц, генерируемых сервером, в зависимости от того, какие запросы формирует пользователь. В настоящий момент для формирования интерфейсных страниц используется технология ASP компании MicroSoft. ASP технологии поддерживаются IIS (Internet Information Server), в состав которой входят: WEB-сервер, SMTP-, NNTP-сервера и ряд дополнительных служб. Эти службы стали стандартными компонентами ОС.

Автор сценария, проектировщик или разработчик приложений доступа к БД и файлам может использовать следующие функциональные возможности:

а) Выполнение сценариев, включенных в веб-страницы. При помощи ASP-страниц можно внедрять сценарии в страницы HTML и применять серверные компоненты ActivX, чтобы реализовать динамическую бизнес-логику на базе WEB. Сценарии могут быть написаны на языках VB, ScriptingEdition, Jscript, а также на любом другом языке создания сценариев ActivX, для которого имеется соответствующая поддержка IIS.

б) Доступ к БД. Можно сделать при помощи Microsoft Data Access Components (MDAC)-набора методов БД, интегрированных с IIS.

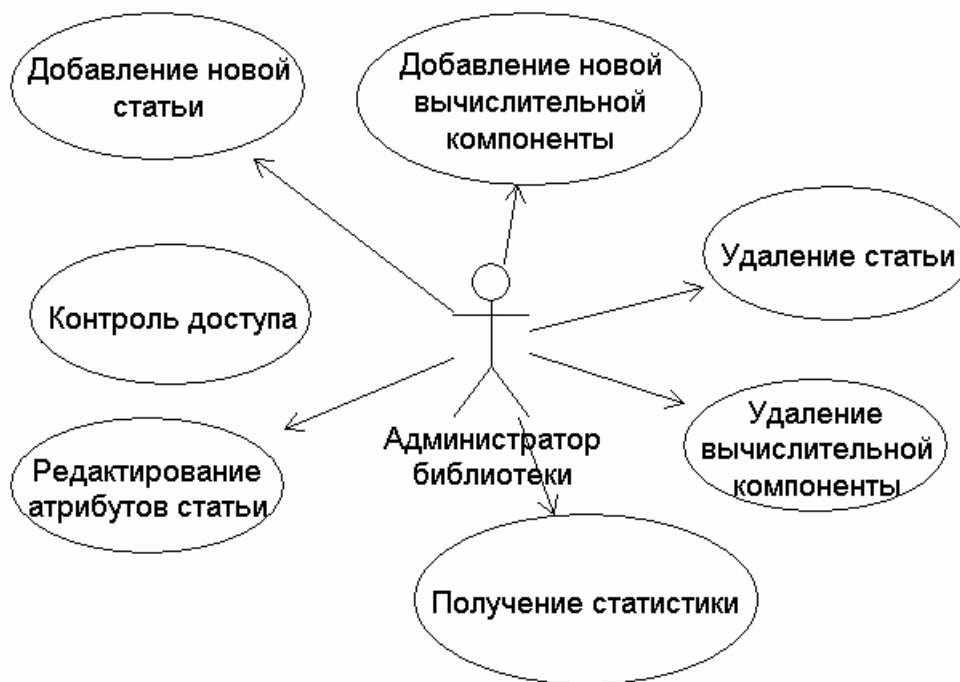


Рис 1. Варианты использования утилит поддержки базы данных

Для создания сложных составных документов (например, WORD, в который внедрен EXCEL) разработана и реализована OLE-технология. Она позволяет внедрять и связывать объекты (при изменении одного меняется и содержание другого). При дальнейшем усовершенствовании появилась компонентная модель объектов COM, независимая от языка создания, но совместимая во взаимодействии.

COM определяет двоичный интерфейс, полностью независимый от языка программирования. Следующим шагом явилась автоматизация объектов OLE (Object Linking and Embedding), которая обеспечивает доступ к объектам приложения и манипуляцию с ними извне. Автоматные объекты доступны только программно и временно. Два типа автоматных серверов: внутри процесса и вне процесса. Сервер внутри процесса является DLL.

Задачи, решаемые при помощи DCOM:

- создание кода, который может использоваться многими языками программирования;
- создание элементов управления ActivX;
- управление другими программами через автоматизацию OLE;
- взаимодействие с объектами или программами на других компьютерах DCOM, который позволяет создать объект в одном приложении или DLL, а затем вызвать методы этого объекта из приложения, которое работает на другом компьютере.

Ниже (рис 2) приведены основные прецеденты, реализованные в интерфейсе пользователя использующие HTTP-страницы и DCOM-программы. Все функциональные компоненты, кроме получения результатов вычислений, выполняются в виде соответствующих запросов к базе данных. В настоящей версии получение результатов расчетов осуществляется через FTP протокол. Адрес ресурса посылается пользователю по электронной почте по окончанию расчетов.

Все процедуры поиска выполняются через интерфейс с базой данных на основе SQL запросов. В данной версии интерфейс осуществляется через MS SQL Server и компоненты интерфейса с базами данных, поставляемые с IIS (Internet Information Services).

Вычислительные компоненты представляют собой тандем, состоящий из ASP файла, формирующего входные данные для проведения вычислений, и собственно вычислительной компоненты, разработанной в технологии ActiveX.

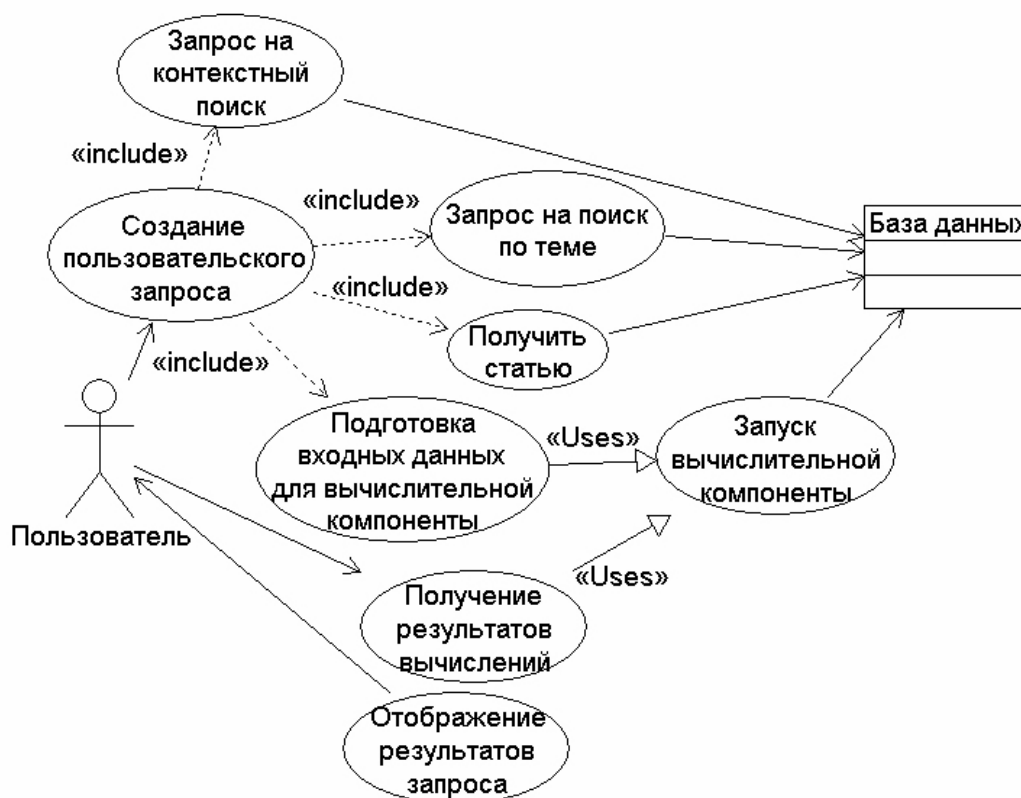


Рис 2. Варианты использования для интерфейса пользователя.

3. Структура базы данных

В данной статье мы рассмотрим минимальную структуру данных, необходимую для функционирования интерактивной библиотеки алгоритмов и программ. Собственно эта структура использовалась для создания макетной версии библиотеки. База данных была реализована с использованием реляционной СУБД MS ACCESS. Данная база данных была выбрана из соображений простоты и удобства разработки SQL запросов. Структура базы данных не учитывает никаких особенностей, присущих только MS ACCESS.

Для реализации минимальных требований, изображенных на рис. 2, достаточно двух связанных таблиц: таблицы статей и таблицы вычислительных модулей.

Как показано на рис. 3, таблицы связаны по ключевому полю таблицы статей.

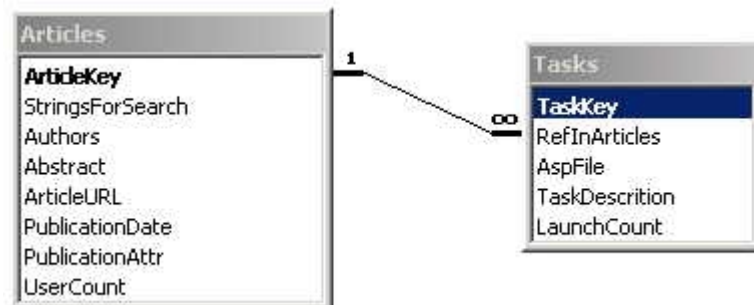


Рис 3. Общая структура базы данных

Сначала рассмотрим структуру таблицы со статьями «Articles». Таблица состоит из одного ключевого поля, шести текстовых полей, одного поля типа «Дата» и одного числового поля. Рассмотрим назначение полей подробнее.

Имя поля	Тип данных	Описание
ArticleKey	Счетчик	Уникальный ключ статьи
StringsForSearch	Текстовый	Набор строк для контекстного поиска, разделенных запятой, размер 255 символов.
Authors	Текстовый	Авторы статьи. Размер 255 символов
Abstract	Текстовый	Краткое содержание на 255 символов
ArticleURL	Текстовый	URL (путь) файла с текстом статьи. Размер 255 символов. Файл в формате ASP или HTML.
PublicationDate	Дата/время	Дата первой публикации статьи
PublicationAttr	Текстовый	Библиографическая ссылка на печатное издание. Размер 255 символов.
UserCount	Числовой	Количество пользователей, затребовавших статью.
SubjectStrings	Текстовый	Чему посвящена статья. Размер 255 символов.

Рис. 4. Структура таблицы статей

- Поле ArticleKey – уникальный ключ, используемый для связи с таблицей расчетных модулей.
- Поле StringsForSearch – используется в запросах на контекстный поиск.
- Поле Authors используется для идентификации статьи в списке найденных статей по контекстному поиску или на заданную тему, а также формирования списка статей заданного автора.
- Поле Abstract используется для идентификации статьи в результирующем списке найденных статей.
- Поле ArticleURL служит для формирования ссылки в результирующем HTML файле на файл со статьей в формате HTML.
- Поля PublicationDate и PublicationAttr также помещаются в итоговый список найденных статей для того, чтобы пользователь мог выбрать наиболее свежую статью, или статью, прошедшую отбор в авторитетный журнал.
- Поле UserCount используется в статистических отчетах для выявления наиболее популярных статей.
- В поле SubjectString находятся данные, используемы в запросах на поиск по заданной теме.

В данной структуре кажутся пересекающимися по назначению поля StringForSearch и SubjectStrings, но мы считаем, что в поле для контекстного поиска логично поместить очень короткие фразы, имеющие отношение к теме статьи, в последнее поле лучше записывать хорошо сформулированное название предметной области, к которой относится данная статья. В макетной версии не используется поле с ссылкой на файл с исходными текстами примеров применения алгоритмов, описываемых в статье, в окончательную версию такое поле будет включено.

Таблица расчетных модулей имеет ключевое поле, два текстовых поля и два числовых.

Имя поля	Тип данных	Описание
TaskKey	Счетчик	Уникальный ключ расчетного модуля
RefInArticles	Числовой	Ссылка на запись в таблице статей, к которой относится данный расчетный модуль.
AspFile	Текстовый	URL ASP файла для запуска расчетного модуля. Размер 255 символов.
TaskDescription	Текстовый	Краткое описание расчетного модуля. Размер 255 символов.
LaunchCount	Числовой	Счетчик количества запусков расчетного модуля.

Рис. 5. Структура таблицы с расчетными модулями

- Поле TaskKey в данной версии не используется, но в дальнейшем будет использоваться для формирования таблицы перекрестных ссылок между расчетными модулями, относящимися к одной теме, но разным статьям. Это можно делать для того, чтобы облегчить пользователю сравнение и выбор алгоритмов.
- Поле RefInArticles служит для обработки запроса на формирование списка расчетных модулей, относящихся к одной статье.
- Поле AspFile – служит для формирования ссылки в файле с текстом статьи, по которой будет осуществляться переход к формированию исходных данных для расчетов и запуск собственно расчетного модуля.
- Поле TaskDescription также используется для формирования ссылки на переход к расчетам, для того, чтобы пользователь мог сделать осознанный выбор.
- Поле LaunchCount предполагается использовать в статистических отчетах.

В окончательной версии библиотеки предполагается завести таблицу с отзывами пользователей о той или иной статье. Мы здесь не приводим структуры SQL запросов, так их описание занимает много места. Но уже из структуры таблиц и содержания предыдущего раздела (рис. 2) можно видеть насколько разнообразные запросы можно предложить пользователю.

Если разработчики расчетных модулей не захотят давать доступ всем желающим, то надо добавить таблицу со списком допущенных пользователей и их атрибутами

4. Архитектурные особенности библиотеки

Архитектурно библиотека состоит из трех частей, как и любая другая современная система, реализованная в технологии клиент-сервер.

- Пользовательская сторона.
- Сервер, реализующий службы библиотеки.
- База данных

Для того, чтобы пользователь мог получить доступ к ресурсам библиотеки, все что ему необходимо – это компьютер, подключенный к глобальной сети (или локальной сети, где расположен сервер с библиотекой) и интерпретатор HTML файлов версии не ниже 4. На рисунке 6 изображена общая архитектура интерактивной библиотеки с указанием технологий, которые использует соответствующая компонента.

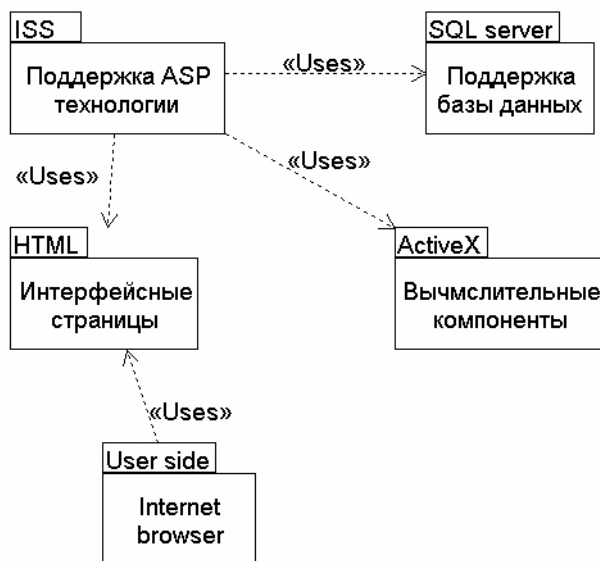


Рис. 6. Общая структура интерактивной библиотеки

Общий процесс взаимодействия можно изобразить, как следующую последовательность действий.

- Пользователь выходит на главную страницу библиотеки.
- Выбирает нужный ему тип запроса.
- Формирует запрос и запускает его на выполнение.
- Выбирает нужную статью.
- При необходимости, запускает нужный расчетный модуль.
- По получении извещения по электронной почте о завершении расчета, копирует результаты расчетов.
- Интерпретирует результаты расчетов своими средствами, в дальнейшем, предполагается разработка компоненты обработки и представления результатов.
- Получает исходные тексты с примерами применения алгоритмов.

Собственно HTML файлы со статьями могут содержать любой тип информации, наиболее наглядно иллюстрирующий содержание.

На наш взгляд, создание подобных информационных ресурсов поможет не только студентам и аспирантам в изучении новых материалов, но и сократит путь внедрения новых алгоритмов в промышленные приложения.

Мир един и целостен и при его изучении методами анализа и синтеза, дифференцирования и интегрирования важно соблюдать меру. Мы дифференцировались, пришла пора интегрироваться. Пусть на страницах Интернета оживут все работы наших сотрудников и это будет достойной памятью трудов Н.Н. Яненко.

5. Список литературы.

- [1] Windows 2000. Санкт-Петербург: БХВ, 2000, 1056 с.
- [2] Хилаер С., Мизик Д. Программирование Active Server Pages. М.: Русская редакция, 1999, 296 с.
- [3] Гололобова С.П. Обзор математических программных систем. В сб. Труды ИВМиМГ №6, Новосибирск, 1998, с. 20-27.
- [4] Билл Гейтс «ДОРОГА В БУДУЩЕЕ». М.: Русская редакция, 1996, 312с.
- [5] Буч Г., Рамбо Д., Джекобсон А. UML–Руководство пользователя М.: ДМК, 2000, 412 с.