

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «КубГУ»)

Факультет компьютерных технологий и прикладной математики
Кафедра математического моделирования

КУРСОВАЯ РАБОТА

РАЗРАБОТКА СУБД С ИСПОЛЬЗОВАНИЕМ XML-ФАЙЛОВ

Работу выполнил _____ А.М. Пасевич
(подпись)

Направление подготовки 01.03.02 Прикладная математика и информатика

Направленность (профиль) «Математическое моделирование и
вычислительная математика» (Математическое моделирование)

Научный руководитель
канд. физ.-мат. наук, доц. _____ М.С. Капустин
(подпись)

Нормоконтролер
канд. физ.-мат. наук, доц. _____ С.Е. Рубцов
(подпись)

Краснодар
2018

РЕФЕРАТ

Курсовая работа 19 стр., 3 рис., 6 ист.

БАЗА ДАННЫХ, СУБД, XML, ДАННЫЕ, XML-ФАЙЛЫ.

Объектом исследования курсовой работы являются СУБД, базы данных и их связь с XML-файлами.

Цель курсовой работы – разработка СУБД с использованием XML-файлов. В курсовой работе представлено графическое изображение СУБД в виде схемы, которая позволяет в дальнейшем организовать целостный программный продукт.

СОДЕРЖАНИЕ

Введение.....	4
1 Теоретические аспекты СУБД.....	5
1.1 Основные понятия БД.....	5
1.2 Модели данных.....	5
2 База данных.....	7
2.1 Разновидности СУБД и их обеспечение безопасности.....	8
2.1.1 Многопользовательские СУБД.....	8
2.1.2 Персональные СУБД.....	8
3 Функции СУБД.....	11
3.1 Определение данных.....	11
3.2 Обработка данных.....	11
3.3 Организация безопасной сохранности данных.....	12
3.4 Словарь данных.....	12
3.5 Производительность.....	13
4 Основные понятия языка XML.....	14
4.1 Определение XML.....	14
4.2 XML и база данных.....	15
4.3 Обоснование выбора языка программирования.....	16
Заключение.....	18
Список использованных источников.....	19

ВВЕДЕНИЕ

Развитие средств вычислительной техники обеспечило для создания и широкого использования систем обработки данных разнообразного назначения. Разрабатываются информационные системы для обслуживания различных систем деятельности, систем управления хозяйственными и техническими объектами, модельные комплексы для научных исследований, системы автоматизации проектирования и производства, всевозможные тренажеры и обучающие системы. Одной из важных предпосылок создания таких систем стала возможность оснащения их «памятью» для накопления, хранения и систематизация больших объемов данных. Другой существенной предпосылкой нужно признать разработку подходов, а также создание программных и технических средств конструирования систем, предназначенных для коллективного пользования. В этой связи потребовалось разработать специальные методы и механизмы управления такого рода совместно используемыми ресурсами данных, которые стали называться базами данных. Исследования и разработки, связанные с проектированием, созданием и эксплуатации баз данных, а также необходимых для этих целей языковых и программных инструментальных средств привели к появлению самостоятельной ветви информатики, получившей название системы управления данными.

Такие программные комплексы выполняют довольно сложный набор функций, связанный с централизованными управлениями, данными в базе данных интерфейсах всей совокупности ее пользователей. По существу, система управления базами данных служит посредником между пользователями и базой данных.

В настоящее время разработаны и используются на персональных компьютерах около двадцати систем управления базами данных. Они представляют пользователю удобные средства интерактивного взаимодействия с БД и имеют развитый язык программирования.

1 Теоретические аспекты СУБД

1.1 Основные понятия БД

Всякая прикладная программа является отображением какой - то части реального мира и поэтому содержит его формализованное описание в виде данных. Крупные массивы данных размещают, как правило, отдельно от исполняемой программы, и организуют в виде Базы данных. Начиная с 60-х годов для работы с данными, стали использовать особые программные комплексы, называемые системами управления базами данных (СУБД). Системы управления базами данных отвечают за:

- физическое размещение данных и их описаний;
- поиск данных;
- поддержание баз данных в актуальном состоянии;
- защиту данных от некорректных обновлений и несанкционированного доступа;
- обслуживание одновременных запросов к данным от нескольких пользователей (прикладных программ). [1]

1.2 Модели данных

Хранимые в базе данных имеют определенную логическую структуру, то есть, представлены некоторой моделью, поддерживаемой СУБД. К числу важнейших относятся следующие модели данных:

- иерархическая;
- сетевая;
- реляционная;
- объектно-ориентированная;

В иерархической модели данные представляются в виде древовидной (иерархической) структуры. Она удобна для работы с иерархически

упорядоченной информацией и громоздка для информации со сложными логическими связями.

Сетевая модель означает представление данных в виде произвольного графа. Достоинством сетевой и иерархической моделей данных является возможность их эффективной реализации показателей затрат памяти и оперативности. Недостатком сетевой модели данных является высокая сложность и жесткость схемы БД, построенной на ее основе.

Реляционная модель данных (РМД) название получила от английского термина Relation - отношение. Модель данных описывает некоторый набор родовых понятий и признаков, которыми должны обладать все конкретные СУБД и управляемые ими БД, если они основываются на этой модели.

Объектно-ориентировочная модель данных – это когда в базе хранятся не только данные, но и методы их обработки в виде программного кода. Это перспективное направление, пока также не получившее активного распространения из-за сложности создания и применения подобных СУБД. [1]

2 База данных

Это совокупность записей различного типа, содержащая перекрестные ссылки.

Файл – это совокупность записей одного типа, в котором перекрестные ссылки отсутствуют.

Более того, в определении нет упоминания о компьютерной архитектуре. Дело в том, что, хотя в большинстве случаев БД действительно представляет собой один или (чаще) несколько файлов, физическая их организация существенно отличается от логической. Таблицы могут храниться как в отдельных файлах, так и все вместе. И, наоборот, для хранения одной таблицы иногда используются несколько файлов. Для поддержки перекрестных ссылок и быстрого поиска обычно выделяются дополнительные специальные файлы.

Поэтому при работе с базами данных обычно применяются понятия более высокого логического уровня: запись и таблица, без углубления в подробности их физической структуры. [1]

На рисунке 1 показан пример базы данных, созданной на СУБД Microsoft Access.

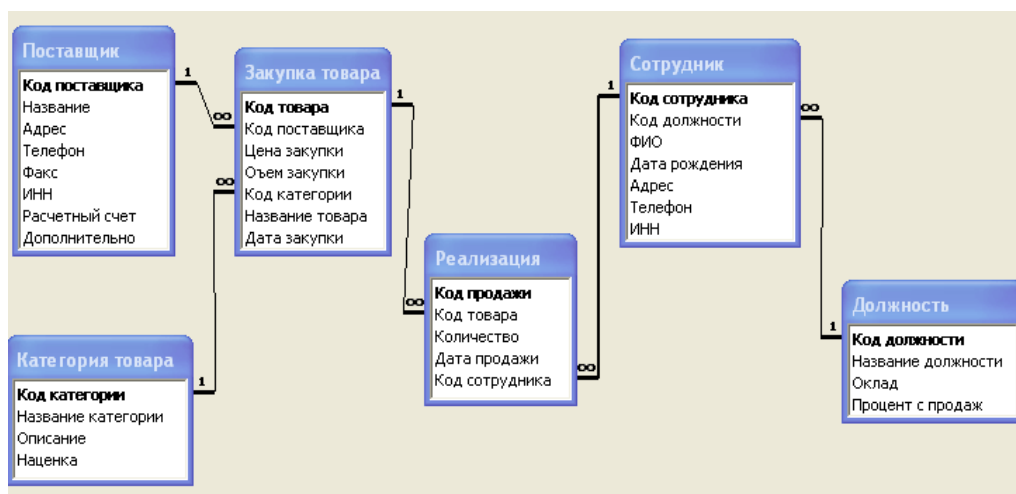


Рисунок 1– Схема данных Базы данных «Мебельный магазин»

Таким образом, сама по себе база данных – это только набор таблиц с перекрестными ссылками. Чтобы универсальным способом извлекать из нее группы записей, обрабатывать их, изменять и удалять, требуются специальные программы, называются СУБД.

2.1 Разновидности СУБД и их обеспечение безопасности

По характеру использования СУБД делят на персональные (СУБДП) и многопользовательские (СУБДМ).

2.1.1 Многопользовательские СУБД

К многопользовательским СУБД относятся, например, СУБД ORACLE и INFORMIX. Многопользовательские СУБД включают в себя сервер БД и клиентскую часть, работают в неоднородной вычислительной среде допускаются разные типы ЭВМ и различные операционные системы. Поэтому на базе СУБДМ можно создать информационную систему, функционирующую по технологии клиент-сервер. Универсальность многопользовательских СУБД отражается соответственно на высокой цене и компьютерных ресурсах, требуемых для поддержки. [2]

2.1.2 Персональные СУБД

К персональным СУБД относятся VISUAL FOXPRO, ACCESS и др. СУБДП представляет собой совокупность языковых и программных средств, предназначенных для создания, ведения и использования БД. [2]

На рисунке 2 изображена персональная СУБД Microsoft Access и создание на ней базы данных.

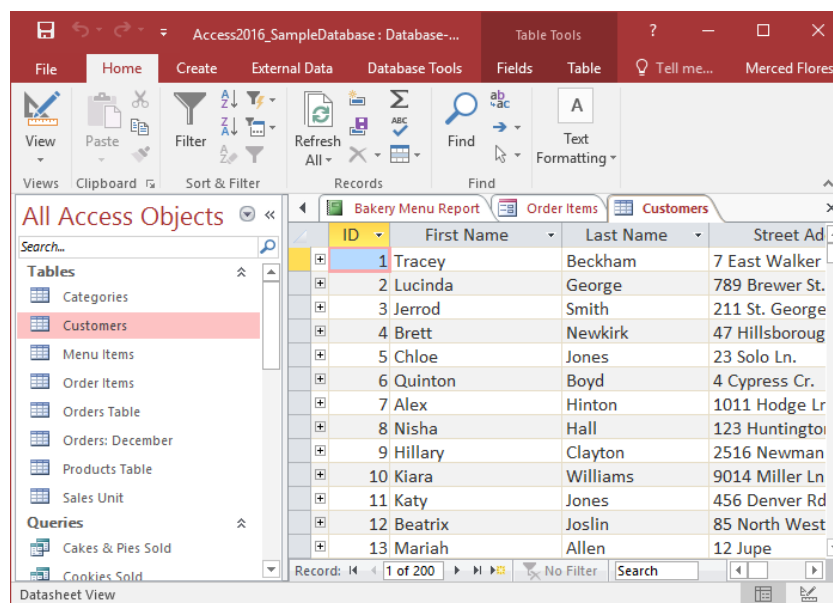


Рисунок 2– СУБД Microsoft Access

Персональные СУБД обеспечивают возможность создания персональных БД и недорогих приложений, работающих с ними, и при необходимости создания приложений, работающих с сервером БД.

Для обработки команд пользователя или операторов программ в СУБДП используются интерпретаторы команд (операторов) и компиляторы.

С помощью компиляторов в ряде СУБДП можно получать исполняемые автономно приложения– exe-программы.

Обеспечение целостности БД-необходимое условие успешного функционирования БД. Целостность БД-свойство БД, означающее, что база данных содержит полную и непротиворечивую информацию. Для обеспечения целостности БД накладывают ограничения целостности в виде некоторых условий, которым должны удовлетворять хранимые в базе данные. Примером таких условий может служить ограничение диапазонов возможных значений атрибутов объектов, сведения о которых хранятся в БД, или отсутствие повторяющихся записей в таблицах реляционных БД. [2]

Обеспечение безопасности достигается СУБД шифрованием прикладных программ, данных, защиты паролем, поддержкой уровней доступа к базе данных, к отдельной таблице.

Расширение возможностей пользователя СУБДП достигается за счет подключения систем распространения Си или Ассемблера.

Поддержка функционирования в сети обеспечивается:

- средствами управления доступом пользователей к совместно используемым данным, т.е. средствами блокировки файлов (таблиц), записей, полей, которые в разной степени реализованы в разных СУБДП;
- средствами механизма транзакций, обеспечивающими целостность БД при функционировании в сети.

3 Функции СУБД

3.1 Определение данных

СУБД должна допускать определения данных (внешние схемы, концептуальную схему, внутреннюю схему, а также все связанные отображения) в исходной форме и преобразовывать эти определения в форму соответствующих объектов. Иначе говоря, СУБД должна включать в себя компонент языкового процессора для различных языков определений данных. СУБД должно также «понимать» синтаксис языка определений данных. [2]

3.2 Обработка данных

СУБД должна уметь обрабатывать запросы пользователя на выборку, изменение или удаление существующих данных в базе данных или на добавление новых данных в базу данных. Другими словами, СУБД должна включать в себя компонент процессора языка обработки данных.

Запросы языка обработки данных бывают «планируемые» и «не планируемые».

Планируемый запрос—это запрос, необходимость которого предусмотрена заранее. Администратор базы данных, возможно, должен настроить физический проект БД таким образом, чтобы гарантировать достаточное быстродействие для таких запросов.

Не планируемый запрос—это, наоборот, специальный запрос, необходимость которого не была предусмотрена заранее. Физический проект БД может подходить, а может и не подходить для рассматриваемого специального запроса. В общем, получение возможной наибольшей производительности для не планируемых запросов представляет собой одну из проблем СУБД. [2]

3.3 Организация безопасной сохранности данных

СУБД должна контролировать пользовательские запросы и пресекать попытки нарушения правил безопасности и целостности, определенные АБД.

СУБД или другой связанный с ней программный компонент, обычно называемый администратором транзакций, должны осуществлять необходимый контроль над восстановлением данных и дублированием. [2]

3.4 Словарь данных

СУБД должна обеспечить функцию словаря данных. Сам словарь данных можно по праву считать БД (но не пользовательской, а системой). Словарь «содержит данные о данных» (иногда называемые метаданными), т.е. определения других объектов системы, а не просто «сырые данные». В частности, исходная и объектная формы различных схем (внешних, концептуальных и т.д.) и отображений будут сохранены в словаре. Расширенный словарь будет включать также перекрестные ссылки, показывающие, например, какие из программ какую часть БД используют, какие отчеты требуются тем или иным пользователям, какие терминалы подключены к системе и т.д. Словарь может быть (а на самом деле даже должен быть) интегрирован в определяемую им БД, а значит, должен содержать описание самого себя. Конечно, должно быть возможность обращения к словарю, как и к другой БД, например, для того узнать, какие программы и/или пользователи будут затронуты при предполагаемом внесении изменения в систему. [2]

3.5 Производительность

Очевидно, что СУБД должна выполнять все указанные функции с максимально возможной эффективностью.

Подводя итог сказанному, можно сделать вывод, что в целом назначением СУБД является предоставление пользовательского интерфейса с БД. Пользовательский интерфейс может быть определен как граница в системе, ниже которой все невидимо для пользователя. Следовательно, по определению пользовательский интерфейс находится на внешнем уровне. Тем не менее, иногда встречаются случаи, когда внешнее представление вряд ли значительно отличается от относящейся, по мере в современных коммерческих продуктах.

В заключении вкратце сопоставим описанную СУБД с системой файлами (или с управлением файлами). В своей основе система управления файлами является компонентом общей системы, которая управляет хранимыми файлами; проще говоря, она «ближе к диску», чем СУБД. Таким образом, пользователь системы управления файлами может создавать и уничтожать хранимые файлы, а также выполнять простые операции выборки и обновления хранимых записей в таких файлах. Однако, в отличие от СУБД, системы управления файлами имеют некоторые недостатки. [2]

4 Основные понятия языка XML

4.1 Определение XML

XML (англ. eXtensible Markup Language, расширяемый язык разметки) представляет собой язык разметки, который был создан для описания данных. Язык разметки представляет собой набор символов или последовательностей, вставляемых в текст для передачи информации о его выводе или строении. Таким образом, текстовый документ, размеченный с помощью такого языка, содержит не только сам текст, но и дополнительную информацию о его структуре. Кроме того, язык разметки позволяет вставлять в документ интерактивные элементы и содержание других документов. Разметка разделяется на стилистическую разметку, структурную и семантическую: стилистическая разметка отвечает за внешний вид документа, структурная разметка задает структуру документа, семантическая позволяет описать логику представления данных. XML представляет собой подмножество метаязыка SGML, разработанное для упрощения процесса машинного разбора документа. Можно сказать, что он сам является метаязыком, т.к. не ограничивается набором определенных тегов и используется в качестве средства для описания грамматики других языков и контроля за правильностью составления документов. XML-документ обычно состоит из процессинговых инструкций, элементов, атрибутов, сущностей и комментариев. [3],[4],[6]

Целью создания XML было обеспечение совместимости при передаче структурированных данных между разными системами обработки информации, особенно при передаче таких данных через Интернет. Словари, основанные на XML (например, RDF, RSS, MathML, XHTML, SVG), сами по себе формально описаны, что позволяет программно изменять и проверять документы на основе этих словарей, не зная их семантики, то есть не зная

смыслового значения элементов. Важной особенностью XML также является применение так называемых пространств имён (англ. namespace). [3]

4.2 XML и базы данных

Многие веб-сайты используют XML в качестве хранилищ данных, информация из которых затем преобразуется в HTML или XHTML для вывода пользователям. Растет и многообразие исходных данных. Например, новое поколение продуктов и технологий, таких как Microsoft InfoPath и W3C XForms, также начинают работать с XML данными, перенося их из различных форм непосредственно в такие хранилища данных, как Microsoft Access или SQL Server. [6]

Реляционные базы данных — один из наиболее популярных способов хранения данных. Они давно и тщательно проработаны, хорошо подходят для хранения структурированных данных, хранят огромные количества существующей информации и хорошо знакомы большому количеству разработчиков. Эти причины делают СУРБД хорошими кандидатами на совместное использование с XML, и у вас есть масса возможностей в этом убедиться. Получение XML из реляционных баз данных большое количество HTML и XHTML веб-сайтов используют, непосредственно или опосредованно, реляционные данные. В этой области широко применяются комбинации PHP с базой данных MySQL или ASP или ASP.NET с SQL Server или Microsoft Access. Данные обычно хранятся как реляционные таблицы, и программист пишет код для создания HTML или XHTML, иногда с применением XML в качестве промежуточной стадии. XHTML представляет собой язык приложения XML. Создание XHTML веб-страниц из реляционных данных демонстрирует один из способов, которыми эти данные могут использоваться для получения XML-кода для представления данных. Распространенность этой технологии является наилучшим доказательством

возможности отображения реляционных данных на иерархические структуры XML и передачи этих структур пользователю. [6]

4.3 Обоснование выбора языка программирования

Главной задачей разработки является создание программы, реализующей систему управления базой данных. В качестве языка программирования для разработки приложения был выбран PHP. Преимущества PHP заключаются в том, что он отлично работает вместе с другими языками и сервисами вроде XML, HTML и различными базами данных. Скрипты PHP можно легко смешивать и вставить между тегами HTML, что позволяет контенту быть весьма динамичным. Благодаря этому, функция и куски кода могут быть записаны в любом порядке внутри документа. [4],[5]

Соответственно, PHP будет использоваться для написания скриптов серверной части СУБД. XML-файлы будут использоваться для определения структуры базы данных и хранения в них необходимой информации. Обращение к серверу СУБД будет реализовано посредством модификации SQL через Web-клиент, написанный на PHP и HTML. Структура будет выглядеть как приведенная ниже схема. [4], [6]

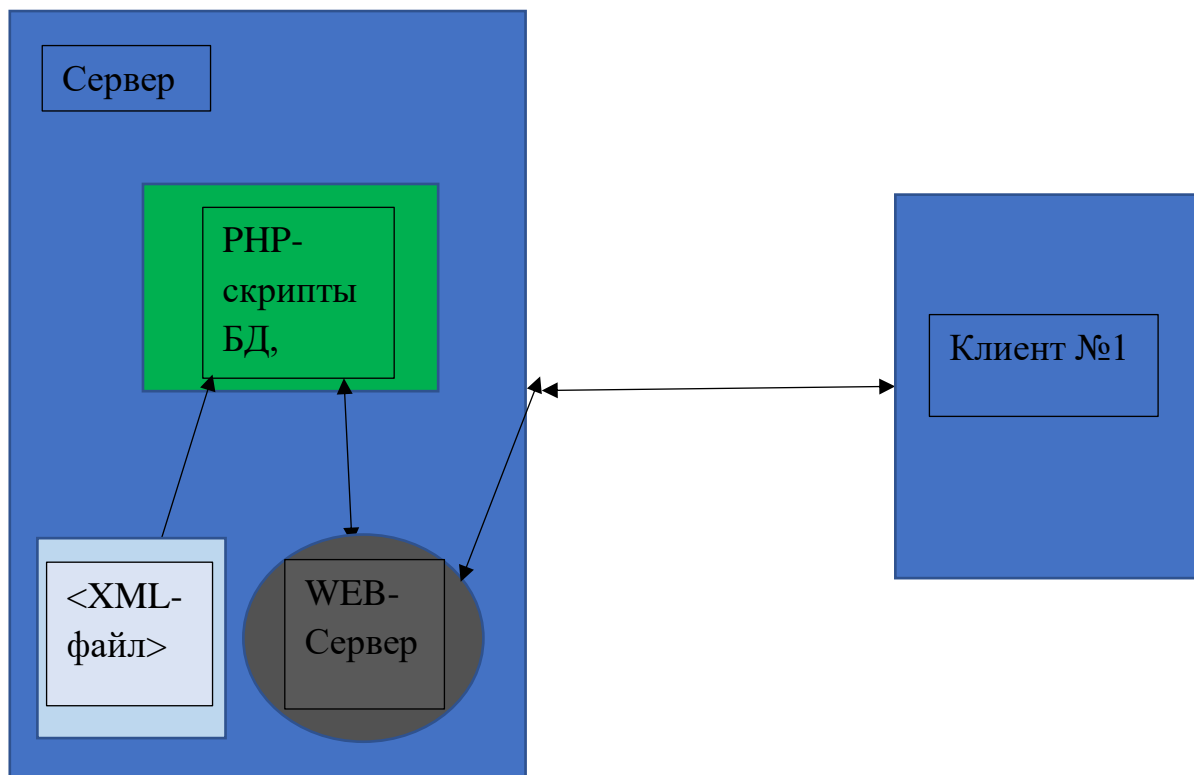


Рисунок 3—Схема примерной организации работы программы.

ЗАКЛЮЧЕНИЕ

Цель курсовой работы – разработка СУБД с использованием XML-файлов.

В рамках курсовой работы были рассмотрены теоретические основы как создания баз данных и СУБД в целом, так и использования XML-файлов .

Были изучены функции, разновидности СУБД, а также возможности хранения, редактирования и использования данных из XML-файлов. Также, для составления примерной схемы организации работы программы были изучены аспекты языка программирования PHP, а также языка разметки XML, способы их взаимодействия в сфере разработки СУБД. В дальнейшем, используя полученные знания, можно будет разработать гибкую СУБД, которая сможет быть использована, как прикладная программа в большинстве проектов.

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

- 1 Кузнецов, С. Д. Основы баз данных, 2-е изд. / С.Д. Кузнецов, М., Интернет-университет информационных технологий; БИНОМ. Лаборатория знаний, 2007. — 484 с.
- 2 Уидом, Дж. Системы баз данных. Полный курс./ Г. Гарсиа-Молина, Дж. Ульман, Дж. Уидом,: ВИЛЬЯМС, 2003. – 1088с.
- 3 Одиночкина, С.В. Основы технологий XML. Учебное пособие./ С.В. Одиночкина, -СПб,: НИУ ИТМО, 2013. – 56с.
- 4 XML. Работа с XML, 4-е издание/Д. Хантер и др., -М.: ДИАЛЕКТИКА, 2009 – 1344с.
- 5 Шлоснейгл, Дж. Профессиональное программирование на PHP/ Дж.Шлоснейгл, -М.: ВИЛЬЯМС, 2006 – 413с.
- 6 XML и Базы данных [Электронный ресурс] URL:
<http://www.dialektika.com/PDF/978-5-8459-1533-7/part.pdf> (14.12.2018)