

Федеральное государственное автономное
образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»
ИНСТИТУТ ПЕДАГОГИКИ, ПСИХОЛОГИИ И СОЦИОЛОГИИ
КАФЕДРА СОВРЕМЕННЫХ ОБРАЗОВАТЕЛЬНЫХ ТЕХНОЛОГИЙ

УТВЕРЖДАЮ
Заведующий кафедрой
_____ И.А. Ковалевич
подпись инициалы, фамилия
« _____ » _____ 2016 г.

БАКАЛАВРСКАЯ РАБОТА

09.03.03.19 Прикладная информатика в социальных коммуникациях

Разработка базы данных сотрудников предприятия ООО «Мостовик-Восток»

Научный руководитель _____ доцент, к.и.н. _____ О.М. Долидович
подпись, дата должность, ученая степень инициалы, фамилия

Выпускник _____ _____ Н.С. Мальчикова
подпись, дата инициалы, фамилия

Красноярск 2016

РЕФЕРАТ

Выпускная квалификационная работа по теме «Разработка базы данных сотрудников предприятия ООО «Мостовик-Восток»» содержит 52 страниц текста, 1 приложение, 18 использованных источников, 1 таблицу, 15 рисунков.

БАЗА ДАННЫХ, СИСТЕМА УПРАВЛЕНИЯ БАЗАМИ ДАННЫХ, ПРОМЫШЛЕННОЕ ПРЕДПРИЯТИЕ, РАЗРАБОТКА БАЗ ДАННЫХ.

Целью дипломной работы является разработка базы данных сотрудников предприятия Общество с ограниченной ответственностью «Мостовик-Восток».

Объект исследования: базы данных.

Предмет исследования: база данных с информацией о сотрудниках предприятия ООО «Мостовик-Восток».

Дипломная работа состоит из двух основных частей.

В первой части рассматриваются теоретические аспекты разработки баз данных и их использования на промышленных предприятиях. Изучаются виды систем управления базами данных, описываются их характеристики.

Во второй части диплома описывается направление предприятия ООО «Мостовик-Восток». Так же дано обоснование и описание выбора среды программирование и программного пакета MicrosoftAccess, термины и особенности данного приложения.

В результате была разработана база данных с информацией о сотрудниках предприятия ООО «Мостовик-Восток». База имеет таблицы с информацией о всех сотрудниках, запрос на заработную плату, отчет о телефонах и адресах проживания сотрудников.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	4
Глава 1 Теоретические аспекты разработки баз данных промышленных предприятий.....	7
1.1 Использование баз данных на промышленных предприятиях.....	7
1.2 Виды и современные системы управления базами данных.....	15
1.2.1 Характеристика реляционных СУБД.....	18
1.2.2 Характеристика объектных СУБД.....	21
1.2.3 Характеристика распределенных СУБД.....	25
1.2.4 PostgreSQL.....	27
1.2.5 IBM DB2.....	29
1.2.6 SQL Server.....	32
Глава 2 Разработка базы данных сотрудников предприятия ООО «Мостовик-Восток».....	36
2.1 Направление деятельности предприятия ООО «Мостовик-Восток».....	36
2.2 Обоснование разработки базы данных.....	37
2.3 обоснование среды программирования.....	38
2.4 Программный пакет для создания баз данных MicrosoftAccess.....	39
2.5 Структура базы данных сотрудников предприятия ООО «Мостовик-Восток».....	42
ЗАКЛЮЧЕНИЕ.....	47
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ.....	46
ПРИЛОЖЕНИЕ А.....	50

ВВЕДЕНИЕ

В процессе своей деятельности промышленные предприятия, корпорации, ведомственные структуры, органы государственной власти и управления накопили большие объемы данных. Они содержат огромные возможности по извлечению полезной аналитической информации, на основе которой можно выявлять скрытые тенденции, строить стратегию развития, принимать и находить новые решения.

Информационные технологии прочно вошли в современную жизнь. Без них существование и возрастающее развитие предприятий становится невозможным. Широкое распространение получили автоматизированные информационные системы: информационно-справочные, информационно-поисковые. Все они предназначены для регистрации, хранения и обработки данных с целью поиска и выдачи отчетов с информацией пользователям. В большинстве случаев автоматизированные информационные системы разрабатывают как базы данных. При помощи баз данных возможно вести учет товаров в супермаркетах и на складах, начислять зарплату в бухгалтериях, вести учет личного состава предприятия и т.д. Базы данных применяются везде, их использование значительно упрощает операции хранения, поиска, добавления и изменения информации.

Большие объемы и сложность потоков оперативной и статистической информации при планировании, учете и контроле функционирования производства предполагают использование интегрированных банков данных в составе автоматизированных систем обработки данных для организационно-административного управления, а также методов и средств автоматизации решения задач управления в промышленности и на транспорте.

История развития баз данных начинается с 1955 г., с момента появления оборудования для обработки записей. Программное обеспечение этого времени поддерживало обработку записей на основе файлов. Данные

хранились на перфокартах. Оперативные сетевые базы данных появились в 1960-х гг. Операции над такими базами проводились в интерактивном режиме с помощью терминалов. В 1970 г., благодаря работам Эдгора Ф. Кодда, появились реляционные модели данных. Сам же термин «база данных» появился в начале 1960-х гг., но свое широкое употребление он получил в 1970 г.

Базы данных обычно используются в деятельности человека, где нужно работать с большим количеством информации. Первые базы данных стали применяться в таких науках, как ядерная физика, химия, космонавтика.

По мере того, как развивалась компьютерная технология, базы данных стали разрабатываться во всех сферах деятельности человека и использоваться от сельского хозяйства до промышленных предприятий, в школах, институтах, экономических и юридических фирмах, и от телевидения до интернета. Интернет является в настоящее время самой большой базой данных.

На любом предприятии, большом или маленьком, нужно организовывать работу сотрудников так, чтобы это было выгодно и эффективно. Это решается по средством внедрения баз данных на предприятия, что способствует более точному выполнению работ.

Целью дипломной работы является разработка базы данных сотрудников предприятия Общество с ограниченной ответственностью «Мостовик-Восток».

Объект исследования: базы данных.

Предмет исследования: база данных с информацией о сотрудниках предприятия ООО «Мостовик-Восток».

Задачи:

1. Проанализировать виды и современные системы управления базами данных
2. Охарактеризовать способы использования баз данных на промышленных предприятиях

3. Описать направления деятельности предприятия ООО «Мостовик-Восток»
4. Обосновать необходимость разработки базы данных
5. Разработать базу данных.

Глава 1. Теоретические аспекты разработки баз данных промышленных предприятий

1.1 Использование баз данных на промышленных предприятиях

При анализе особенностей функционирования промышленных предприятий, подтверждается необходимость всесторонних исследований и пересмотра действующих бизнес-процессов, подходов и методов к процессу проектирования информационных систем (ИС) промышленных предприятий, а также подходов и методов к проектированию баз данных (БД), как основы любой информационной системы.

Любое крупное предприятие (холдинг, производственный концерн, проектная организация или просто производственное предприятие) имеет дело с огромными объемами информации, которую необходимо обрабатывать и анализировать с целью последующего использования в технологическом цикле производства или бизнес-процессах.

Решение подобного рода задач невозможно осуществить без использования современных информационных систем интеграции, обработки и управления данными. Их применение способствует повышению эффективности работы предприятий, сокращению издержек, оптимизации численности персонала. Разработка таких систем на крупных предприятиях часто связана с необходимостью производить интеграцию данных из высоконагруженных информационных систем и обеспечивать приемлемое время выполнения запроса пользователя [5].

В связи со значительной стоимостью создания такого рода систем и специфичностью деятельности каждой организации, разработка высоконагруженных информационных систем выполняется под заказ, они опираются на имеющиеся готовые решения крупных производителей. Эти системы используются для решения определённых задач и требований, установленных техническими заданиями и практически всегда представляют

собой уникальные продукты, которые на других предприятиях неэффективны или вообще невозможны в использовании.

Фирмы, занимающиеся разработкой информационных систем, не проводят исследования для создания универсального подхода к проектированию потому, что это будет уменьшать коммерческую выгоду от их деятельности.

Нынешняя теория и практика показывают, что эффективность управления деятельностью промышленных предприятий должна быть основана на постоянном совершенствовании структуры базы данных для промышленных предприятий. В данной ситуации особую актуальность приобретает вопрос рефакторинга баз данных как метода радикальных преобразований устоявшейся структуры базы данных промышленных предприятий [1].

Начиная с 1960-х годов, для работы с данными прибегают к использованию особых программных комплексов, называемых системами управления базами данных (СУБД). К функциям систем управления базами данных относятся физическое размещение данных и их описание; 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).[4]:

Модели данных находящиеся в базах имеют определенную логическую структуру, то есть, представлены некоторой моделью, поддерживаемой СУБД. К числу важнейших относятся следующие модели данных: иерархическая; сетевая; реляционная; объектно – ориентированная.

Иерархическая модель одна из самых старых и первых систем управления базами данных. Она представляет собой вид древовидной структуры и удобна в работе с упорядоченной информацией, но имеет трудности в работе, где информация содержит сложные логические связи [15].

Основными информационными единицами в иерархической модели данных являются сегмент и поле.

Сетевая модель – логическая модель данных, являющаяся расширением иерархической модели. Она представляет собой вид произвольного интерфейса. Эффективная реализация оперативности и показателя затрат памяти, являются плюсами иерархической и сетевой моделей данных. К минусам сетевой модели данных относится высокая сложность и жесткость схемы построенной на ее основе базы данных.

Реляционная модель данных получила название от английского термина Relation - отношение. Данная модель описывает некоторый набор признаков и родовых понятий, которыми должны обладать все конкретные СУБД и базы данных, которые ими управляются, если они основываются на этой модели.

Объектно-ориентированная модель данных хранит в базе не только данные, но и методы их обработки в виде программного кода. Из-за сложности создания и применения подобных СУБД, такое направление является нераспространенным [14].

База данных представляет собой один или (чаще) несколько файлов и их физическая организация сильно отличается от логической. Таблицы могут храниться как в отдельных файлах, так и все вместе. И, наоборот, для хранения одной таблицы иногда используются несколько файлов. Так же существуют специальные файлы, которые помогают поддерживать перекрестные ссылки и выполнять быстрый поиск информации [1].

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

Таким образом, база данных представляет из себя набор таблиц с перекрестными ссылками. Чтобы работать с базой данных: добавлять, обрабатывать, изменять и удалять данные в ней, требуются специальные программы, называются СУБД.

Главной задачей проектирования архитектуры базы данных, является работа с разными типами и источниками информации. Таким образом, источники и потребители информации служат подразделением данного предприятия, головная контора холдинга, аппарат министерства и предприятия других отраслей (возможные поставщики и потребители, государственные регламентирующие органы и др.). Масштабное объединение бизнеса показывает, что источники и потребители информации могут находиться в любой географической точке, где это будет необходимо.

Отсюда следуют стратегические для архитектуры баз данных и информационных систем в целом решения. Объединение требований к динамике и разнообразию типов информационных потоков, обрабатываемых в информационной системе, дают возможность обобщать характеристику технологий, формирующих архитектуру базы данных в составе информационной системы:

- компонентная технология проектирования и перекомпоновки предметно-ориентированных операционных баз данных, допускающих работу пользователей через общие, в том числе, для хранилища данных, интерфейсы;
- расширенная технология хранилища данных, интегрирующая исторические форматированные данные, архивные текстовые документы, звуковые и видеоархивы, а также картографические данные, и включает в себя средства оперативной аналитической обработки данных, необходимые виды "дружественных" интерфейсов, например: гипертекстовый;
- открытость базы данных для включения в нее и получения из нее информации с использованием принципов глобальной информационной магистрали [17].

Архитектура открытых систем, расширенная методами и средствами компонентного формирования: на верхнем уровне это открытость компонентного проектирования баз данных и свободного обмена с

источниками информации любых внешних систем, на нижнем уровне - технологическая открытость баз данных на основе стандартов переносимости, интероперабельности, масштабируемости и др.

Расширенная технология интегрального хранилища заставляет на новой основе ставить вопрос о разработке интегрированной совокупности интерфейсов пользователя, которая создавала бы естественные условия для работы с информацией и функциями вне зависимости от того, к какому классу хранимых данных разработчик вынужден отнести сегодня его (пользователя) информацию.

В информационных системах новые архитектуры СУБД станут определяющим, но не единственным компонентом интегрирующего программного обеспечения (в том числе - промежуточного). Мониторы транзакций и процессов, средства семантического моделирования и использования понятийных моделей, СУБД- независимые средства разработки и исполнения приложений - другие классы компонентов программного обеспечения, обеспечивающие достижение этой цели [14].

Рекомендуется сохранение независимости от СУБД на основе использования стандартов и инструментов, охватывающих различные СУБД. Отказ от связи с одной СУБД, открытость CASE- репозитория, возможность развития метамodelей, поддерживаемых в репозитории, и применяемых к ним проектных процедур - это лишь минимальные требования к методам и инструментам.

Разумно ориентироваться на CASE-системы, ориентированные на возможность параллельного проектирования компонентов независимыми разработчиками (в том числе - без использования данной CASE-системы) с последующей интеграцией метаинформации.

Средства разработки приложений должны удовлетворять требованиям мобильности приложений и, одновременно, работы в среде распределенной базы данных [1].

Дальнейшее развитие автоматизированных систем управления (АСУ) и автоматизированных систем обработки данных различного назначения, стало началом развития новой информационной технологии, которая создавалась благодаря применению методов и средств искусственного интеллекта, таких, как экспертные системы и системы поддержки принятия решений. Такая информационная технология, с помощью интеллектуального интерфейса, будет улучшать эффективность использования баз данных. Модель интегрированной информационной среды промышленного предприятия (Рисунок 1).

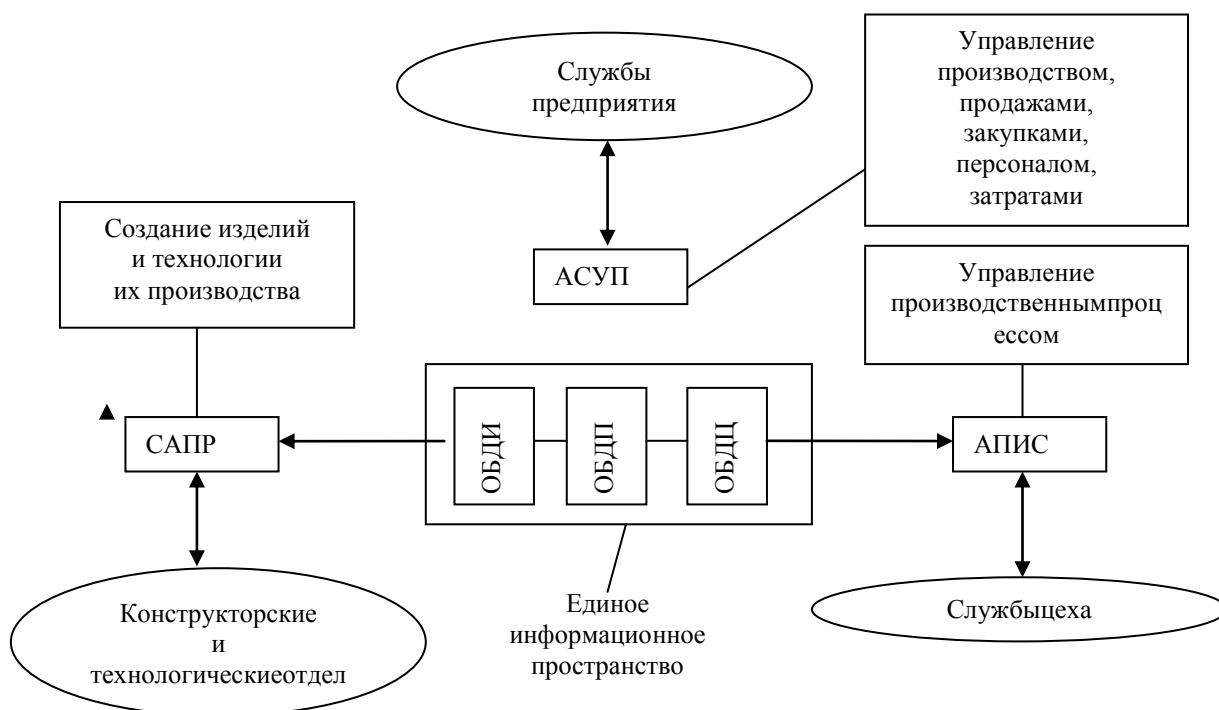


Рисунок 1 - Модель интегрированной информационной среды промышленного предприятия

Кроме информационных ресурсов в интегрированную информационную среду на предприятии входят также следующие компоненты [1]:

- 1) модель управления информационными потоками на предприятии;

- 2) система управления базами данных и системное программное обеспечение;
- 3) набор программных продуктов, автоматизирующих управление информационными потоками;
- 4) телекоммуникационная среда, коммуникационное программное обеспечение, средства организации коллективной работы сотрудников;
- 5) организационная инфраструктура, обеспечивающая функционирование и развитие информационной среды.

Базы данных промышленных предприятий позволяют выполнять задачи по автоматизации управления записями, которые связаны с задачами планирования производства, поставок необходимых ресурсов и сбыта готовой продукции. Такой подход позволяет корректировать плановые задания в процессе производства в зависимости от меняющихся условий: непоставки ресурсов, отказа оборудования, изменения заказа на готовую продукцию и т.п. [5].

Модель интегрированной информационной среды промышленного предприятия включает в себя следующие три главных компонента:

- 1) систему автоматизированного проектирования, с помощью которой конструкторские и технологические отделы разрабатывают новые изделия и технологии их производства;
- 2) автоматизированную систему управления предприятием, с помощью которой службы предприятия осуществляют управление производством, продажами, закупками, персоналом;
- 3) автоматизированную производственную исполнительную систему в цехе, с помощью которой руководители и службы управляют производственным процессом [13].

Системная информационная поддержка этих компонентов осуществляется в едином информационном пространстве, представляющем собой совокупность трех распределенных баз данных:

1) общие базы данных об изделиях, содержащие нормативную информацию и справочные документы, используемые при проектировании новых изделий и создании технологии их производства, информацию о технических решениях, признанных типовыми и пригодными для дальнейшего использования, а также сведения об изделиях, находящихся в процессе производства;

2) общие базы данных о предприятии, содержащие нормативную, справочную, плановую и учетную информацию, которая будет использоваться различными службами в их производственной деятельности;

3) общие базы данных о цехе, включающие информацию о номенклатуре деталей и применяемости их на изделия, нормах и нормативах, технологических процессах, технологических маршрутах, оборудовании, инструменте, оснастке, транспорте, кадрах, а также плановые и учетные данные, показатели деятельности цеха, производственного участка, бригады.

Разъединенная информационная среда представляет собой совокупность баз данных, содержащих сведения, создаваемые и используемые всеми структурными подразделениями предприятия в процессе их производственной деятельности [1].

В составе общей базы данных о предприятии выделены следующие шесть разделов:

- маркетинг;
- экономика и финансы;
- производство;
- снабжение;
- кадры;
- система качества.

Информация в этих разделах формируется и используется в процессе маркетинговых исследований, экономической, финансовой и

производственной деятельности, а также в управлении персоналом и качеством продукции.

1.2 Виды и современные системы управления базами данных

Сегодня установка и администрирование баз данных - это гораздо менее сложный процесс, чем еще несколько лет назад. Проектирование и управление базой данных в большой степени автоматизированы. Программное обеспечение, позволяющее решить эту задачу - создавать базу данных, обновлять хранящуюся в ней информацию - и обеспечивающее быстрый и удобный доступ к ней с целью просмотра и поиска информации, называется системой управления базами данных.

Система управления базами данных создает на экране компьютера определенную среду для работы пользователя (пользовательский интерфейс). Кроме того, СУБД имеет определенные режимы работы и систему команд. На основе СУБД создаются и функционируют информационные системы. Так же, системы управления базами данных - это одна из самых успешных технологий во всей компьютерной отрасли. Системы управления базами данных играют исключительную роль в организации современных промышленных, инструментальных и исследовательских информационных систем.

Характерными режимами работы с СУБД являются создание, редактирование, управление и поиск в базе данных. Для работы в каждом режиме существует своя система команд СУБД. Любая работа пользователя с базой данных строится в форме алгоритма, составленного из этих команд. Такие алгоритмы могут выполняться в режиме прямого выполнения (отданная команда сразу выполняется) и в режиме автоматического выполнения, т. е. в программном режиме [7].

На сегодняшний день существует достаточно много СУБД различного предназначения. В современных информационных системах стандартом де-факто являются реляционные СУБД, данные в которых хранятся в таблицах. Все эти системы имеют свои отличия. Перечислим ключевые параметры,

которые являются важными как для пользователей, так и для разработчиков системы.

Способности к БД:

- Файл-серверные СУБД
- Клиент-серверные СУБД
- Встраиваемые СУБД

В файл-серверных СУБД (MicrosoftAccess, Paradox, FoxPro, dBase и др.) для приложений открыт общий доступ ко всем файлам базы данных (которые обычно хранятся в каком-нибудь разделяемом файловом хранилище) и они могут совместно обрабатывать эти данные. Каждое приложение обрабатывает данные самостоятельно. На сегодня файл-серверная технология считается устаревшей, и её использование в крупных информационных системах является недостатком. Суть проблемы заключается в том, что у файл-серверных СУБД отсутствуют многие преимущества клиент-серверных, таких как: параллелизм запросов, кэширование данных, высокая производительность, и, в тоже время, им присущ ряд недостатков (сложности восстановлением, блокировками, с поддержанием целостности базы и т.д.), что в свою очередь приводит к понижению производительности и надёжности. В файловых СУБД необходимо постоянно отслеживать состояние базы и время от времени осуществлять её «лечение» посредством встроенных или внешних утилит [10].

В клиент-серверных СУБД (InterBase, MicrosoftSQLServer, Firebird, Oracle, MySQL, PostgreSQL и др.) вся обработка данных производится в одном месте – на сервере, там же, где обычно хранятся данные, при этом доступ к файлам данных имеет только один сервер, одна система – сама система управления базами данных. При этом приложения-клиенты лишь посылают запросы к СУБД на обработку и получение данных и получают ответы; непосредственного доступа к файлам данных приложения-клиенты не имеют.

На данный момент практически все промышленные СУБД являются именно клиент-серверными.

Встраиваемые СУБД (MicrosoftSQLServerCompact, FirebirdEmbedded, SQLite, и др.) поставляются в составе какого-нибудь готового программного продукта, для них требуется самостоятельная установка. Задачей СУБД этого типа является локальное хранение данных приложения. На коллективное использование в сети они не рассчитаны. Например, одна из встраиваемых бесплатных СУБД SQLite широко используется в мобильной ОС Android и во многих мобильных приложениях [11].

По схеме лицензирования СУБД бывают:

- Бесплатные;
- Коммерческие промышленные (в большинстве случаев наличием бесплатной ограниченной версии).

Практически все файл-серверные и встраиваемые СУБД являются бесплатными, наиболее известные из бесплатных клиент-серверных СУБД: PostgreSQL, Firebird и MySQL.

Многие производители промышленных СУБД предоставляют возможность пользоваться бесплатными редакциями своих продуктов, которые имеют ограничения по производительности и по функциям в отличие от полнофункциональной версии СУБД.

К плюсам свободных СУБД можно отнести:

- они бесплатны;
- не требовательны к ресурсам персонального компьютера;
- при правильной настройке обладают широкими функциями и большой производительностью;
- надёжны.

Минусы: нет гарантии, что через определенное время проект не прекратится, поскольку он поддерживается сообществом любителей, также

букдет непросто найти знающего специалиста для работы с СУБД типа PostgreSQL или Firebird [10].

Преимуществами коммерческих СУБД являются:

- высокая производительность;
- надёжность;
- масштабность;
- поддерживаемость;
- хорошая задокументированность;
- наличие встроенных инструментов для администрирования и разработки.

И такие компании как, Microsoft, IBM или Oracle, не прекратят поддержку своих системы.

Минусами являются: высокая требовательность к ресурсам, чем у бесплатных аналогов и высокая стоимость.

В основе классификации СУБД лежит используемая модель баз данных, это позволяет выделить несколько классов СУБД: сетевые, иерархические, реляционные, объектные и др. Есть и такие СУБД, которые поддерживают несколько моделей данных одновременно. Более ранние СУБД, такие как сетевые и иерархические имеют древовидную структуру и принцип построения "Предок - потомок". Но такие системы считаются устаревшими и ими почти перестали пользоваться. Им на смену пришли реляционные СУБД [14].

1.2.1. Характеристика реляционных СУБД

Еще в 1970-х годах были получены первые теоретические разработки в области реляционных СУБД и в то же время появились первые прототипы реляционных СУБД. Долгое время невозможно было добиться эффективной реализации таких систем. Но, постепенное накопление методов и алгоритмов

организации реляционных баз данных и управления ими привели к тому, что уже в середине 80-х годов ранние СУБД были вытеснили с мирового рынка реляционными системами. В реляционном подходе организации СУБД предполагается наличие набора отношений (двумерных таблиц), которые являются связанными между собой. Связь в данном случае - это ассоциирование двух или более отношений (таблиц). База данных обладающая ограниченной структурой и не имеющая связей между таблицами - не может называться реляционной. Запросы к таким базам данных возвращают таблицу, которая повторно может участвовать в следующем запросе. Данные в одних таблицах связаны с данными других таблиц, откуда и произошло название "реляционные" [11].

Реляционный подход в построении СУБД имеет ряд преимуществ:

- небольшой набор абстракций, которые позволяют легко моделировать большую часть распространенных предметных областей и допускают точные формальные определения, оставаясь понятными;
- наличие простого и в то же время мощного математического аппарата, опирающегося главным образом на теорию множеств и математическую логику и обеспечивающего теоретический базис реляционного подхода к организации баз данных;
- возможность ненавигационного манипулирования данными без необходимости знания конкретной физической организации баз данных во внешней памяти.

Реляционная модель имеет строгое теоретическое обоснование. Для определения и управления реляционными базами данных был создан декларативный язык SQL. Другие сильные стороны реляционной модели - простота, пригодность для систем интерактивной обработки транзакций, обеспечение независимости от данных. Однако реляционная модель данных и реляционная СУБД, в частности, имеют и некоторые минусы [11].

Главным минусом реляционных СУБД считается ограниченность использования в областях, где достаточно сложные структуры данных.

Единственность и неделимость данных - это основные аспекты традиционной реляционной модели данных, которые хранятся на пересечении строк и столбцов таблицы. Такое правило было заложено в основу реляционной алгебры при ее разработке как математической модели данных. Кроме того, специфика реализации реляционной модели не позволяет адекватно отражать реальные связи между объектами в описываемой предметной области. Данные ограничения существенно мешают эффективной реализации современных приложений, которые требуют уже несколько иных подходов к организации данных.

Главный принцип реляционной модели - удалять повторяющиеся поля и группы с помощью процесса, который называется нормализацией. Плоские нормализованные таблицы универсальны, просты в понимании и теоретически достаточны для показа данных любой предметной области. Они хорошо подходят для приложений, связанных с хранением и отображением данных в традиционных отраслях, таких как банковские или учетные системы, но их использование в системах, основанных на более сложных структурах данных, бывают затруднительными. В основном, это связано с простотой механизмов хранения данных, лежащих в основе реляционной модели [6].

ORACLE, Informix, IBM (DB2), Sybase, Microsoft, Progress и другие – это известные в настоящее время фирмы производители реляционных СУБД. Программы производителей СУБД рассчитаны на работу на различных типах компьютеров (от майнфреймов до портативных) и на различных операционных системах. Также производители СУБД обратили внимание на программы, работающие на настольных компьютерах, такие как dBase, FoxPro, Access и им подобные. Данные СУБД предназначены для работы на PC и решают локальные задачи на одном PC или небольшой группе PC. Часто данные СУБД используются, как зеркальное отображения небольшой части общей корпоративной СУБД, для минимизации требуемых аппаратных и ресурсных затрат для решения небольших задач.

Различные СУБД работают под управлением разных операционных систем и аппаратной части. UNIX, VAX, Solaris, Windows – эти известные операционные системы. В зависимости от объема хранения данных, количества пользователей, осуществляющих одновременный доступ к данным, сложности задач - используются различные СУБД на разных платформах. Для решения задач по обеспечению данными сотни тысяч пользователей, подходит СУБД Oracle на Unix - многопроцессорный сервер. В настоящее время наибольший интерес представляют СУБД направленные на операционную систему Windows использующие платформу Intel.

1.2.2 Характеристика объектных СУБД

Появление объектных баз данных было определено необходимостью решать задачи, связанные с обработкой и хранением сложных многосвязных данных, а также слабоструктурированной и неструктурированной информации: текстом, изображениями, музыкой и т.д. Объектная СУБД идеально подходит для описания такого рода данных, в отличие от реляционных СУБД, где добавление нового типа данных достигается ценой потери производительности или за счет резкого увеличения сроков и стоимости разработки приложений.

Объектная технология - это опыт отражения того, как в действительности человек думает об информации и использует ее. Теоретически, сущности реального мира описываются как объекты, имеющие состояние (что представляется текущими значениями данных объектов) и поведение (которое можно наблюдать и на которое можно влиять через программный код объектов) [13].

Объекты, в отличие от реляционных таблиц, тесно связывают данные и программный код. Объект представляет собой пакет, содержащий значения всех данных этого объекта ("свойства") и копию всех его кодов ("методы").

Методы объекта направляют сообщения для взаимодействия с другими методами этого же или других объектов.

В объектной технологии свойства данных не сводятся к простым "компьютерным" типам данных. Построение точных и удобных моделей данных может осуществляться, когда объект содержит в себе другие объекты или ссылки на них.

Объектные СУБД реализуют весь набор функций, характерных системам управления базами данных плюс возможности объектного программирования. Таким образом, мы получаем все преимущества СУБД наряду с мощным объектным языком программирования (среди них C++, Java) объектов базы [14].

В объектной технологии сложность структур данных находится внутри самих объектов, а доступ к информации производится через простой разнообразный интерфейс. Объектное программирование подходит для разработки сложных приложений потому, что объекты позволяют легко и просто моделировать комплексные данные. Аналогично происходит пополнение и изменение базы данных через объектный доступ.

Доступ к различным источникам данных, в том числе и к данным реляционных СУБД можно осуществить с помощью объектной базы данных. Интерфейсами СУБД с объектными языками программирования, являются C++, Java и набор ActiveX-элементов (модулей, воспринимающих высокоуровневые команды от приложений VisualBasic, Delphi и т.д.), которые разработчик может использовать в своей программе для работы с СУБД.

Основными понятиями, с которыми работает эта модель, являются следующие:

- 1) Наследование - порождает один класс объектов из другого. Новый класс (подкласс) сохраняет все свойства и методы своего "родителя", может иметь дополнительные свойства и методы, характерные только для него.

Множественное наследование - подкласс может иметь более одного "родителя".

2) Инкапсуляция дает возможность описывать объект как своеобразный "черный ящик". Независимо от уровня сложности или класса, может иметь определенное число общедоступных свойств и методов. Приложению не обязательно знать, как объект устроен и действует изнутри. Оно взаимодействует только со свойствами и методами объекта.

3) Полиморфизм означает, что методы, принадлежащие различным классам, могут использовать один и тот же интерфейс вне зависимости от конкретной реализации этих методов. При этом каждый объект осуществляет метод так, как это определено для данного класса объектов.

Дополнительно существует еще две особенности объектного подхода - типизация и сохраняемость. Типизация защищает разработчика от некорректного использования в прикладных программах объектов одного класса вместо другого. Сохраняемость (или хранимость) позволяет объекту существовать в системе после завершения процесса, это является важнейшим для концепции объектных СУБД [5].

Сравнивая реляционный подход с объектным, можно выяснить, что в реляционных БД существуют только два принципиально разных класса объектов:

- реляционная таблица с конечным набором операций, которые допустимы для отношений (имеются в виду операции над множествами);
- встроенные процедуры, работающие с отношениями.

Но из этих двух классов объектов нельзя создавать совершенно новые типы ввиду того, что в реляционных базах данных отсутствуют полноценные механизмы характерные для объектного подхода.

Таким образом, можно выделить преимущества объектных СУБД. Выделим несколько из них. Объектные СУБД беспечивают инкапсуляцию логики и данных в одном объекте; поддерживают сложные типы данных и работу на более высоком уровне абстракции, что позволяет с одной стороны

создавать сложные структуры данных, в том числе мультимедийные, а с другой - обеспечить простоту их развития.

Но объектные СУБД также имеют ряд недостатков и ограничений, среди которых в первую очередь следует выделить отсутствие развитых средств выборки и анализа данных, и единой методологии проектирования объектной БД.

Со времен СССР давно и активно развивались объектные СУБД. В этой области известны такие разработки как: GoodBase, ODB-Jupiter, Dss. Данные разработки совершенно различны, выполнялись в разное время и применялись для различных задач (GoodBase - для решения задач в металлургии, ODB-Jupiter - для создания систем хранения и поиска документов, Dss - для создания систем контроля и управления технологическими процессами) [8].

Среди современных программных продуктов-лидеров объектных СУБД, можно отметить: VERSANT (Versant, Inc), ObjectStore (ObjectDesign, Inc), POET (POET Software, Inc), Jasmine (ComputerAssociates, Inc).

Большое значение для создания корпоративных информационных систем и различных прикладных программ, имеет объектная мультимедийная СУБД Jasmine (компания ComputerAssociatesInternatonalInc, совместно с Fujitsu).

Популярность объектных СУБД в настоящее время сильно растет, что связано с широкими возможностями по их применению для построения информационных систем корпоративного уровня.

В конце рассмотрим еще один вид СУБД. Этот вид начал появляться в начале 1990-х годов. В то время рынок объектных СУБД начал быстро набирать обороты. Из-за чего доходы компаний от продаж реляционных СУБД начали падать. Тогдаими была предпринята попытка включить некоторые особенности объектной модели в реляционные СУБД. Так и появились гибридные реляционно-объектные СУБД [10].

Несколько исследователей доказывали, что реляционно-объектное представление данных является следующим шагом в развитии объектной модели. В ходе детального знакомства с такими продуктами, было выяснено, что такой подход будет являться неполноценным. Практика показала, что на базе реляционно-объектных СУБД нельзя создать эффективные прикладные системы. При сравнении реляционного и объектного подходов становится ясно, в чем причина провала такого рода систем [2].

1.2.3 Характеристика распределенных СУБД

Еще одна классификация СУБД строится на методах организации хранения и обработки данных, поэтому их делят на централизованные и распределённые. Централизованная СУБД хранится на одном компьютере, но пользователь работает с базой данных через удаленный доступ (в режиме клиент-сервер). Большинство централизованных СУБД выполняя только свои стандартные функции, которые усложняются за счёт одновременности доступа многих пользователей к данным, а организацию удаленного доступа к данным оставляет на сетевое обеспечение.

Распределенная СУБД - комплекс программ, предназначенный для управления распределенной базой данных и позволяющий сделать распределенность информации "прозрачной" для конечного пользователя. Термин "прозрачность" означает то, что для конечного пользователя должен быть полностью скрыт тот факт, что распределенная база данных состоит из нескольких фрагментов, которые могут размещаться на нескольких компьютерах, расположенных в сети и к ней возможен параллельный доступ нескольких пользователей [14].

Основное назначение распределенной СУБД состоит в обеспечении средств интеграции локальных баз данных, располагающихся в некоторых узлах компьютерной сети, для того, чтобы пользователь, работающий в любом узле сети, имел доступ ко всем этим базам данных как к единой.

Распределенные СУБД обладают абсолютными преимуществами перед централизованными, а именно:

- отражают структуру организации;
- обладают разделяемостью и локальной автономностью;
- обеспечивают высокую доступность данных;
- обладают высокой надежностью и повышенной производительностью.

Распределенные СУБД имеют и недостатки:

- Распределенные СУБД являются более сложными программными комплексами, чем централизованные СУБД, что обусловлено распределенной природой используемых ими данных, а также репликацией данных.
- Увеличение сложности означает и увеличение затрат на приобретение и сопровождение распределенных СУБД.
- В распределенных системах требуется организовать контроль доступа не только к данным, реплицируемым на несколько различных узлов, но и защиту сетевых соединений самих по себе.
- В распределенных СУБД повышенная стоимость передачи и обработки данных может мешать организации эффективной защиты от нарушений целостности данных.
- Отсутствуют стандарты на каналы связи и протоколы доступа к данным, а также отсутствуют инструментальные средства и методологии, способные помочь пользователям в преобразовании централизованных систем в распределенные.
- Еще не накоплен необходимый опыт промышленной эксплуатации распределенных систем, сравнимый с опытом эксплуатации централизованных систем.
- Распределенные СУБД сложны в управлении, что определяет потенциальную опасность потери целостности данных [14].

Наиболее полно функции распределенной СУБД реализованы в системах: INGRES/STAR, разработанная отделением IngresDivision фирмы The ASK GroupInc.; ORACLE фирмы ORACLE Corp.; модуле распределенной системы DB2 фирмы IBM. Наиболее близко подошли к реализации функций распределенных СУБД такие как: InformixOn-line фирмы InformixSoftware; SybaseSystem 10 фирмы SybaseInc.

1.2.4 PostgreSQL

PostgreSQL это высокопроизводительная объектно-реляционная СУБД с открытыми исходными текстами. Ее разработка продолжается более 15 лет и улучшается архитектура, чем он и завоевала репутацию надежной, масштабной и интегрированной СУБД. Она может запускаться на всех основных платформах, включая Linux, Windows, MacOSX. Она имеет полное соответствие с ACID, полностью поддерживает ключи, объединения, представления, триггеры, и хранимые процедуры (на разных языках). Она имеет API для Java, C/C++, Perl, Ruby, Python, ODBC, Tcl, и многие другие [7].

PostgreSQL является СУБД класса предприятия и предоставляет такие особенности как восстановление по точке во времени, Multi-VersionConcurrencyControl (MVCC), табличное пространство, асинхронную репликацию, вложенные транзакции (точки сохранения), планировщик/оптимизатор запросов, горячее резервирование и упреждающее журналирование на случай сбоя. Он имеет поддержку международных кодировок, в том числе и многобайтовых, при использовании различных кодировок можно различать регистр, использовать полнотекстовый поиск и сортировку. Большое число работающих одновременно пользователей и большое количество подконтрольных данных, тем не менее, не особо влияют на масштабность системы. Существуют действующие PostgreSQL системы,

которые могут управлять более чем 4 терабайтами данных. В таблице 1 представлены некоторые общие ограничения PostgreSQL [5].

Таблица 1 - Перечень ограничений СУБД PostgreSQL

Ограничение	Значение
Максимальный размер базы данных	Неограниченно
Максимальный размер таблицы	32 TB
Максимальный размер строки	1.6 TB
Максимальный размер поля	1 GB
Максимальное кол-во строк в таблице	Неограничено
Максимальное кол-во столбцов в таблице	250 - 1600 в зависимости от типа столбцов
Максимальное кол-во индексов в таблице	Неограничено

В PostgreSQL могут выполняться хранимые процедуры, которые написаны на различных языках программирования, включая C/C++, Java, Perl, Ruby и собственном PL/pgSQL, который аналогичен Oracle'sPL/SQL. В стандартной библиотеке функций имеются сотни встроенных функций - от базовых строковых и математических операций до криптографических функций и функций, которые обеспечивают совместимость с Oracle. Хранимые процедуры и триггеры можно писать на Си и загружать в базу данных в качестве библиотеки, расширяя тем самым ее возможности. Также в PostgreSQL включены средства разработки, которые позволяют создавать пользовательские типы данных вместе с операторами и функциями, описывающими их поведение. Как результат, могут быть созданы и добавлены к системе различные типы данных - от геометрических и пространственных примитивов до специальных типов данных, определенных в ISBN/ISSN [12].

Наряду с языками, которые могут использоваться при написании хранимых процедур, существует и большое количество интерфейсных

библиотек, которые позволяют как компилируемым, так и интерпретируемым языкам взаимодействовать с PostgreSQL. Это интерфейсы для ODBC, Perl, Java (JDBC), C, C++, Lisp, Scheme, PHP, Qt и др.

Также, исходный код PostgreSQL доступен под лицензией BSD – наиболее либеральной из открытых лицензий. Эта лицензия позволяет свободное использование, модификацию и распространение PostgreSQL в любой форме, с закрытым или открытым исходным кодом. Можно поступать со сделанными модификациями так, как вам будет угодно. Таким образом, PostgreSQL является не только мощной системой управления базами данных, которая позволяет обеспечивать деятельность организации, но и платформой разработки для создания приложений, которые требуют использования реляционной СУБД [8].

1.2.5 IBM DB2

DB2 представляет собой семейство систем управления реляционными базами данных, которые выпускает корпорация IBM.

На сегодняшний день СУБД DB2 представлена версиями для следующих платформ:

- DB2 for z/OS v10 для платформ z/OS и OS/390
- DB2 for Linux, UNIX and Windows v10 для платформ HP-UX, AIX, Linux, Windows, Solaris и бета-версия для платформы MacOSX;
- DB2 Server for VSE & VM v7 для платформ z/VSE и z/VM;
- DB2 for i для платформы IBM i (является встроенной в систему на аппаратно-программном уровне).

В IBM DB2 используется общепринятая на данный момент клиент-серверная архитектура реляционной СУБД, с функцией обеспечения хранения информации на сервере и возможностью подключения приложений-клиентов к базам данных через сеть либо локально [2].

В DB2 для конкурентов доступ к данным со стороны параллельно исполняемых приложений обеспечивается транзакционным механизмом, который основан на использовании блокировок и ведения журнала транзакций, о так же обеспечивает предоставление стандартных гарантий ACID.

Также в DB2 осуществлен механизм поддержки всех распространенных промышленных стандартов доступа приложений к данным, включая интерфейсы JDBC и ODBC, стандартный язык запросов SQL, работу с типовыми текстовыми табличными форматами и т.п. К тому же, в DB2 включены развитые возможности хранения и работы с полуструктурированными данными в форматах JSON/BSON и XML. Для разработки хранимых процедур DB2 обладает поддержкой множества процедурных языков, включая:

- Язык SQL/PL, используемый в СУБД Oracle,
- Язык SQLPL, стандартный для DB2,
- Разработка «внешних» хранимых процедур на языках Java, C, C++ и COBOL.

Характерными особенностями DB2 являются:

- Встроенные средства контроля и разграничения доступа, которые реализуют модель мандатного разграничения доступа, а также предоставляют возможности гранулярного ограничения доступа к информации в разрезе объектов (таблиц, представлений);
- Масштабность, ограниченная только имеющимися вычислительными ресурсами, и обеспечивает максимально экономичное и эффективное использование вычислительных ресурсов;
- Наличие развитой интегрированной системы резервного копирования и восстановления данных;
- Полный набор технологий для построения «классических» аналитических хранилищ данных (материализованные представления, деление таблиц на разделы, «внутренняя» параллелизация исполнения

сложных запросов, оптимизация при кэшировании данных и сканировании таблиц и индексов, и т.п.);

- Высокая отказоустойчивость и осуществление линейного масштабирования кластерных конфигураций DB2 pureScale, с возможностью хранения данных на дисках с общим доступом;
- Поддержка механизма построения конфигураций массивно-параллельной аналитической обработки данных (MPP) из большого количества серверов, объединенных в коммуникационной сети, на базе DB2 DPF;
- Поддержка технологии DB2 VLU, реализующей поддержку современной in-memory «поколonoчного» анализа без применения ручной оптимизации структуры баз данных.

Плюсы и минусы данной СУБД.

К плюсам относятся:

- наличие хорошей бесплатной версии без ограничений на размер базы;
- отсутствие в бесплатной версии ограничителя запросов;
- наличие хорошей бесплатной техподдержки;
- в отличие от PostgreSQL, наличие возможности получения платной поддержки производителя, что позволяет ее применение в Enterprise секторе бизнеса;
- лучше, чем PostgreSQL работает с конфигурациями 1С:Предприятие в автоматическом режиме блокировок;
- обладает хорошей производительностью;
- меньше трудностей, связанных с неуникальностью индексов (для решения проблемы рекомендуется временно загружать базы в DB2);
- отсутствие ограничения на 256 таблиц, что позволяет расширять возможности при работе с RLS.

Минусами являются:

- небольшая распространенность;

- небольшое количество специалистов и высокая стоимость их услуг;
- большой размер баз, чем в других СУБД;
- медленная скорость загрузки dt-файла;
- необходима "тонкая" настройка параметров СУБД, есть автоподстройка системы, но она осуществляет полную настройку;
- некоторые сообщения могут неверно обрабатываться платформой, для решения нужно "понижение уровня" логирования ошибок.

1.2.6 SQLServer

Microsoft SQL Server представляет собой систему управления реляционными базами данных. Microsoft SQL Server разработана компанией Microsoft. В качестве языка запросов используется язык стандарта ANSI/ISO Transact-SQL с расширениями. Система работает с базами данных больших и средних размеров.

Чтобы выйти на рынок корпоративных баз данных, компания Microsoft использовала код, основанный на Sybase SQL Server. На этом рынке баз данных конкурировали такие компании как Oracle, IBM и Sybase. Компании Sybase и Microsoft объединили усилия, чтобы выпустить на рынок программу Ashton-Tate, которую назвали SQLServer 1.0 для OS/2. Она была эквивалентом Sybase SQL Server 3.0 для Unix, VMS и многих других [10].

Программа Microsoft SQL Server была выпущена и включена в состав операционной системы в 1992 году, компанией Microsoft. Позднее компания Microsoft старалась быть абсолютным обладателем прав на все версии SQL Server для Windows.

К моменту выхода Windows ОС у Microsoft и Sybase были собственные модели программного продукта и собственный способ маркетинга в осуществлении.

После компании разъединились, и каждая самостоятельно сделала релизы программ. Sybase изменила название продукта на AdaptiveServerEnterprise - это было связано с путаницей Microsoft SQL Server.

Программа SQL Server 7.0 обладает графическим пользовательским интерфейсом и была первым сервером, имеющим такие возможности. Для решения спорных вопросов с компанией Sybase, чтобы избежать нарушения авторских прав, программный код в этой версии был переписан. А уже в ноябре 2005 года была представлена версия SQL Server 2005. Ее запуск происходил одновременно с запуском VisualStudio 2005 [16].

SQL Server – это база данных, наполненная информацией и которая может расширяться без потери скорости операций с записями в многопользовательском режиме. Новые пользователи могут добавляться путем модернизации оборудования. При проведении последнего теста поддерживалось до 4600 пользователей базы данных.

За счет того, что сетевая безопасность связана с самим сервером безопасности - данные максимально защищены от несанкционированного доступа. Поскольку функции безопасности реализованы на уровне пользователя, их доступ к записи данных может быть ограниченным, защищая их тем самым от поиска или модификации, указав доступ на уровне пользовательских привилегий. К тому же, с данными, которые хранятся на отдельном сервере, сервер работает как шлюз, ограничивающий несанкционированный доступ [6].

SQL Server выполняет обработку запросов от пользователей и отправляет им только результаты запроса. Таким образом, по сети передается минимальная информация, что ускоряет время ответа и позволяет устранять узкие места в сети. Это также, делает SQL Server идеальной базой данных для использования в интернет.

Техническое обслуживание SQL Server очень простое и не требует больших знаний. Возможны изменения в структуре данных, а так же резервное копирование во время работы сервера, без остановки [8].

Два основных языка разработки приложений используется для извлечения информации из данных SQL Server - это C++ и VisualBasic. Эти языки являются частью VisualStudioMicrosoft. Покупка приложений, разработанных с помощью этих продуктов, гарантирует, что программное обеспечение будет модернизироваться и развиваться в будущем.

SQL Server является приложением базы данных при работе на новейшие разработки Microsoft. Выбрав Microsoft SQL Sever в качестве базы данных информации для компании, приложение может расширяться и приспосабливаться по мере изменения бизнес-климата.

Эта система управления базами данными, прежде всего, отличается высокой надежностью. Это возможно за счет применения различных базовых технологий, таких как создание отказоустойчивых кластеров, зеркалирования, предоставления разнообразных средств для работы с журналами.

Следующее достоинство Microsoft SQL Server – это ее возможности по масштабности и по высокой производительности. Начиная с редакции SQL Server 2005, в ней появилась функция партиционирования. Эта технология позволяет разбивать большие таблицы на несколько элементов, которые привязаны к разным файл-группам. В результате данные физически размещаются на нескольких жестких дисках, и таким образом операции чтения/записи распределяются. Данная технология совершенно многозначна для пользовательского приложения, и ее работа осуществляется автоматически, по средствам самой СУБД [7].

Для дальнейшего повышения быстродействия здесь использована технология сжатия, которая работает как на уровне записей, так и страниц. Особое внимание в Microsoft SQL Server уделено вопросам безопасности. В системе управления базами данных реализована поддержка современных криптоалгоритмов. В редакцию SQL Server 2008 были добавлены средства шифрования данных. В результате этого данные теперь хранятся на жестком диске только в зашифрованном виде, а их расшифровка производится «на

лету» при чтении. Ключи шифрования могут храниться не только в самой системе управления базами данных, но также и на внешних аппаратных HASP-модулях [16].

Microsoft SQL Server активно применяется сегодня для создания корпоративных систем. Эта система управления базами данных занимает первое место на рынке по числу проданных копий. Терабайтные внедрения на ее основе давно перестали восприниматься как уникальные события. В последнее время при крупных внедрениях Microsoft SQL Server все чаще подступают к петабайтному уровню.

Глава 2 Разработка базы данных сотрудников предприятия

ООО «Мостовик-Восток»

2.1 Направление деятельности предприятия ООО «Мостовик-Восток»

Предприятие ООО Мостовик-Восток было основано и зарегистрировано в Едином государственном реестре юридических лиц в мае 2000 г., в городе Канск Красноярского края. Основной деятельностью предприятия является строительство автомобильных дорог, тоннелей и мостов.

На протяжении нескольких лет предприятие выполняло небольшие заказы по ремонту мостов и дорог. Но в 2010 г предприятие получило заказ на строительство большого автодорожного моста и, начиная с этого времени, предприятие стало выполнять заказы на строительство мостов, дорог и тоннелей.

Предприятие является частным и количество основных (постоянных) сотрудников составляет 20-30 человек, но ввиду того, что основные работы происходят в летнее время, количество сотрудников в этот период времени увеличивается. Из-за этого происходит большая текучесть кадров. Для эффективной деятельности предприятия, его подразделяют на отделы.

К основным отделам данного предприятия относятся:

- Административный отдел;
- Бухгалтерия;
- Отдел кадров;
- Служба безопасности;
- Технический отдел;
- Транспортный отдел;
- Производительный цех;
- Склад материалов и готовых изделий.

Предприятие имеет технику:

- Камазы;
- Грузоподъемные краны;
- Экскаваторы;
- Бурильная установка;
- Машины для перевозки большого груза;
- Тракторы;
- Автобус;
- Служебная машина.

Техника в транспортном отделе используется не только в рабочих целях самого предприятия, но также предоставляется другим компаниям на заказ. Отдел занимается перевозкой материалов и рабочих к месту строительного объекта.

С недавнего времени предприятие стало также заниматься дополнительной деятельностью: изготовлением бетонных, деревянных и металлических изделий. Для этого служит производственный цех, в основном предприятие производит в большом количестве бетонные изделия, так как они используются в строительстве автодорожных мостов и подземных тоннелей. Так же все эти изделия выставляются на продажу или делаются для заказчиков.

2.2 Обоснование разработки базы данных

При знакомстве с системой хранения и обработки информации в отделе кадров на предприятие ООО «Мостовик-Восток», было выяснено, что большая часть информации о сотрудниках хранится на бумажных носителях.

На обработку таких данных уходит много сил и времени, что тормозит деятельность работников отдела кадров. При этом количество информации с каждым разом увеличивается, накапливается и не успевает проработаться, что приводит к проблемам в отделе кадров и конечно на самом предприятии.

Руководителю предприятия было предложено разработать базу данных для хранения всей информации о сотрудниках. Целью является разработать базу данных сотрудников для отдела кадров, которая будет обеспечивать удобное и быстрое добавление, хранение, поиск, изменения и обработку информации.

Руководитель предприятия выдвинул следующие требования к базе данных:

- Производительность;
- Целостность;
- Возможность быстрого поиска, добавления и изменения данных;
- Простота в использовании;
- Безопасность;
- Минимальные затраты.

2.3 Обоснование выбора среды программирования

Для разработки базы данных была выбрана среда MicrosoftAccess 2013, поддерживаемая операционной системой Windows 7 и выше.

В настоящее время большую популярностью имеет реляционная СУБД ACCESS, входящая в состав пакета MicrosoftOffice.

Со средой программирования Access совмещаются остальные приложения MicrosoftOffice: Word, Excel, PowerPoint.

Access – это многопользовательское приложение, в нем создана надежная система защиты от несанкционированного доступа к файлам.

MicrosoftAccess 2013 мощная и сложная система, но для непрофессиональных пользователей, он доступен и прост в использовании.

Понятный интерфейс и простота настройки, удобное и быстрое создания таблиц, форм, запросов, взаимосвязь с другими приложениями

пакета, средства организации работы с базами данных и защита информации – это список основных достоинств данного приложения.

Так же в состав Access входит множество мастеров, конструкторов и надстроек, которые упрощают процесс создания объектов базы данных.

Основные функции СУБД – это описание структуры базы данных, обработка данных и управление ими.

Главными особенностями при выборе MSAccess стали:

- простота и доступность программы;
- приложение включает в себя компоненты для создания баз данных: таблицы, формы, запросы, отчёты, макросы и модули;
- удобная панель инструментов, позволяющая быстро выполнить различные команды с объектами баз данных;
- возможность просматривать и редактировать связанные записи в режиме таблиц.

2.4 Программный пакет для создания базы данных

MicrosoftAccess

MicrosoftAccess — реляционная система управления базами данных, разработанная корпорацией Microsoft. Обладает широким спектром функций, таких как: механизм связанных запросов, установка связи с внешними базами данных и отдельными таблицами. Благодаря наличию встроенного языка VBA, в Access можно создавать код приложений, работающих с базами данных [4].

Базы данных MicrosoftAccessобладают многими достоинствами, разработчики очень часто предпочитают использовать в своей работе именно это приложение. Преимуществами баз данных MicrosoftAccess является то, что в одном файле может находиться большое количество таблиц и индексов. Поэтому с такими базами данных удобнее и легче работать – создавать

резервные копии, переносить на другие персональные компьютеры и т.д. Еще одним плюсом является то, что имена полей в такой базе данных можно задавать русскими буквами.

В составе средств обработки данных Access присутствуют модули и макросы. В отличие от макросов, модули представляют собой более тонкое и мощное средство создания программных расширений и прикладных приложений в среде Access. Эти модули по своим функциональным средствам относятся к таким профессиональным инструментам разработки, как Delphi или PowerBuilder. Вместе с тем, при использовании модулей, от пользователя требуются навыки и квалификация программиста, а также умение использовать основные принципы объектно-ориентированного программирования.

Для программирования в среде Access используется процедурный язык VisualBasic для приложений (VBA- VisualBasicforApplications) в который добавлены объектные расширения и набор элементов SQL. Процесс создания в Access программных расширений предполагает активное и непосредственное использование подходов объектно-ориентированного программирования.

Объектно-ориентированное программирование основывается на идее "упакованной функциональности", суть которой заключается в том, что процесс построения программы опирается на применение фундаментальных сущностей - объектов. Каждый из объектов обладает индивидуальным набором операций и свойств, которые он может выполнять. Реализация взаимодействий между объектами и контроль за ними возлагается на исполняющую среду разработки, в которой пишется код программы. Поэтому работа программиста, применяющего технологию объектно-ориентированного программирования, сводится к созданию объектов с описанием их свойств, а также реакций объекта на те или иные внешние действия [7].

В Access многие из программных объектов совпадают с физическими объектами базы данных, например: таблицы, формы, отчеты. Составные

объекты, которые включают в себя набор более простых объектов, семейством. Кроме "видимых" объектов имеет место и большое количество "скрытых" объектов, управление которыми доступно только из программных приложений.

Access располагает двумя типами модулей: стандартными и модулями класса. В стандартные модули входят функции процедуры, которые можно вызвать из любого окна базы данных. В таких модулях, как правило, содержится программный код, имеющий универсальный характер, и который может применяться в различных местах данного приложения, а также, в различных приложениях и модулях.

Модули класса включают в себя, модули отчетов и модули форм. Они имеют связь с определенным отчетом или формой. Следует заметить, что в ранних версиях Access они представляли единственно возможный инструмент объектно-ориентированного программирования. В этих модулях содержатся процедуры обработки событий, которые запускаются в ответ на их вызов в отчете или форме. Процедуры обработки событий используют, когда необходимо управление поведением отчета или формы и их откликом на некие события, такие, например, как нажатие кнопки.

Наиболее важная область применения в Access технологии объектно-ориентированного программирования заключается в программировании доступа к данным информационного массива. Для решения этой задачи фирма Microsoft разработала специальный интерфейс - DAO (DataAccessObjects) [14].

DataAccessObjects – это объект, предоставляющий интерфейс к любому типу баз данных или механизму хранения. Этот шаблон программирования применим ко многим языкам программирования, большинству программного обеспечения, которые нуждаются в хранении информации.

2.5 Структура базы данных сотрудников предприятия ООО «Мостовик-Восток»

Начиная работать с базой данных сотрудников, пользователю открывается окно содержащие 4 функциональных кнопки, которые позволят пользователю начать работу с базой данных (Рисунок 2).

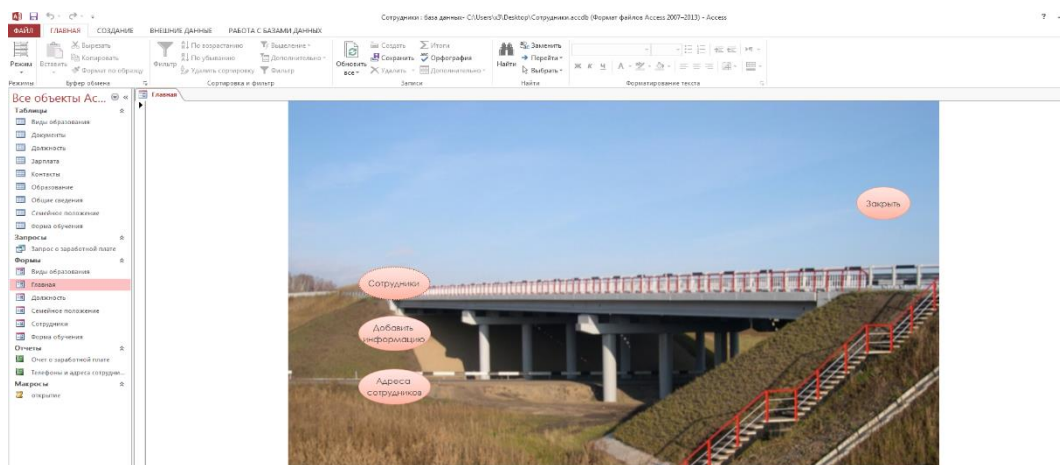


Рисунок 2 – Главная форма базы данных сотрудников предприятия

Первой функциональной кнопкой на главной форме, является «Сотрудники» (Рисунок 3).

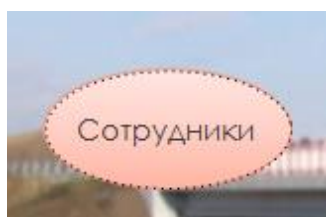


Рисунок 3 –Первая функциональная кнопка «Сотрудники»

При ее нажатии пользователю открывается форма с информацией о сотрудниках (Рисунок 4).

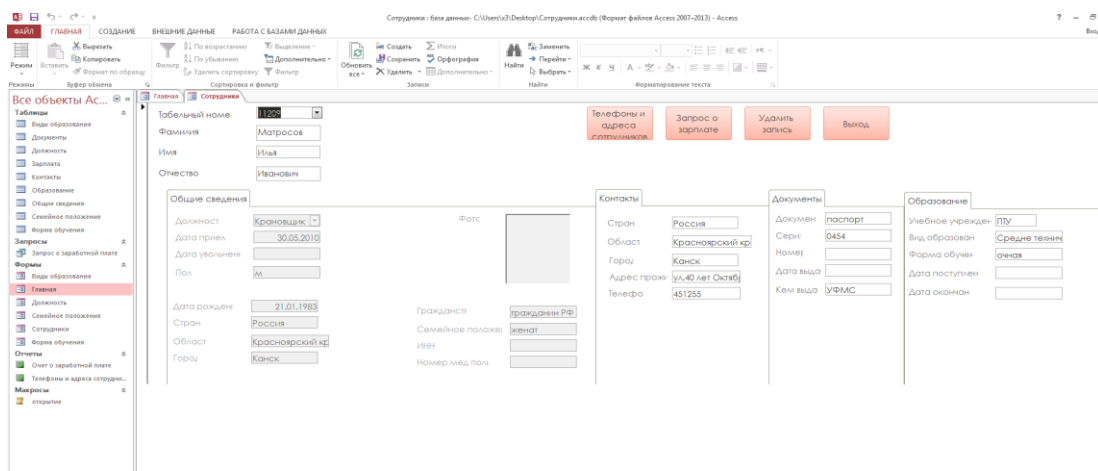


Рисунок 4 – Форма с информацией о сотрудниках

В данном окне пользователь видит всю личную информацию о сотруднике. Форма была создана из нескольких таблиц с данными: «Общие сведения», «Контакты», «Документы», «Образование» (Приложение А).

Форма создана, что бы пользователь не затрачивал много времени для поиска информации о любом сотруднике предприятия. Выбирая из списка «Табельный номер», у каждого сотрудника он индивидуальный, база данных предоставляет всё информацию сотрудника под этим номером.

Так же форма имеет 2 функциональных кнопки: «Удалить запись», «Выход» (Рисунок 5, 6)

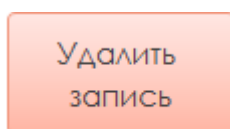


Рисунок 5 – Кнопка удаления записи

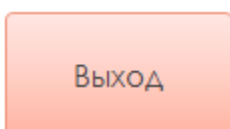


Рисунок 6 – Кнопка выезда из формы с информацией о сотрудниках

Так же если пользователю потребуется информация о заработной плате сотрудников, ее можно получить из запроса, нажав на функциональную кнопку «Запрос о зарплате» (Рисунок 7).

ID	Должность	Фамилия	Имя	Отчество	Оклад
11200	Бригадир	Вейде	Федор	Федорович	9 000,00р.
11204	Бригадир	Захаров	Павел	Сергеевич	9 000,00р.
11212	Бухгалтер	Ненаева	Татьяна	Владимировна	10 000,00р.
11211	Главный бухгалтер	Москвитин	Павел	Андреевич	13 000,00р.
11208	Директор	Малочиков	Сергей	Викторович	35 000,00р.
11205	Заместитель директора	Ковалева	Славислав	Викторович	28 000,00р.
11224	Инженер по технической безопасности	Храмцова	Татьяна	Викторовна	9 000,00р.
11214	Сварщик	Перевалов	Андрей	Сергеевич	9 000,00р.
11215	Сварщик	Полосинский	Юрий	Александрович	9 000,00р.
11232	Сварщик	Щецов	Юрий	Сергеевич	9 000,00р.
11203	Крановщик	Гес	Павел	Петрович	9 000,00р.
11209	Крановщик	Магросов	Илья	Иванович	9 000,00р.
11279	Крановщик	Шумерин	Андрей	Федорович	9 000,00р.
11206	Уборщица производственных помещений	Ковалева	Елена	Николаевна	6 000,00р.
11230	Уборщица производственных помещений	Щапкина	Людмила	Михайловна	6 000,00р.
11210	Сторож	Миронов	Сергей	Петрович	8 000,00р.
11223	Сторож	Фролов	Александр	Александрович	8 000,00р.
11228	Сторож	Шупин	Николай	Николаевич	8 000,00р.
11202	Водитель	Воробьев	Сергей	Константинович	7 000,00р.
11207	Водитель	Латков	Игорь	Игоревич	7 000,00р.
11213	Водитель	Петров	Алексей	Александрович	7 000,00р.
11216	Водитель	Помомарев	Анатолий	Михайлович	7 000,00р.
11218	Эксплуататор	Радичев	Юрий	Павлович	9 000,00р.
11219	Эксплуататор	Соловьев	Дмитрий	Владимирович	5 000,00р.
11217	Бетонщик	Робатов	Сергей	Юрьевич	9 000,00р.
11221	Бетонщик	Тарбеев	Николай	Викторович	9 000,00р.
11226	Бетонщик	Черкасов	Дмитрий	Николаевич	9 000,00р.
11231	Бетонщик	Шилков	Денис	Петрович	9 000,00р.
11222	Монтажник	Тимафеев	Иррилл	Сергеевич	9 000,00р.
11225	Монтажник	Чуханенко	Николай	Михайлович	9 000,00р.
11227	Монтажник	Циплинов	Сергей	Анатолеевич	5 000,00р.
11220	Завхоз	Соколов	Сергей	Иванович	13 000,00р.

Рисунок 7 – Запрос о зарплате

Второй функциональной кнопкой на главной форме, является «Добавить информацию» (Рисунок 8). При её нажатии пользователю открываются 3 основных таблицы: «Общие сведения», «Контакты», «Документы» (Приложение А).

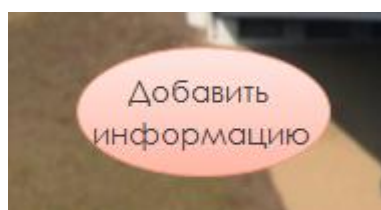


Рисунок 8 – Вторая функциональная кнопка «Добавить информацию»

В данных таблицах пользователь может добавить информацию о новом сотруднике, удалить лишние данные или изменить имеющиеся.

Третьей функциональной кнопкой на главной форме, является «Адреса сотрудников». При её использовании открывается отчет с адресами и телефонами сотрудников (Рисунок 9).

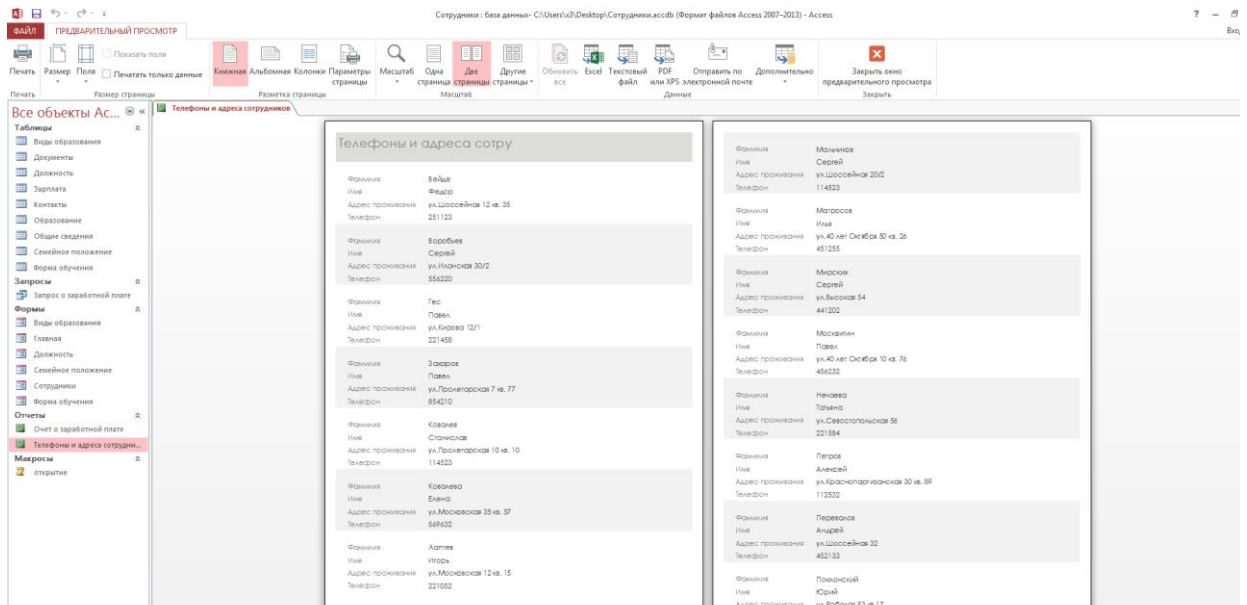


Рисунок 9 – Отчет с адресами и телефонами сотрудников

Четвертой функциональной кнопкой на главной форме, является «Закреть». Это кнопка полностью закрывает базу данных сотрудников (Рисунок 10).

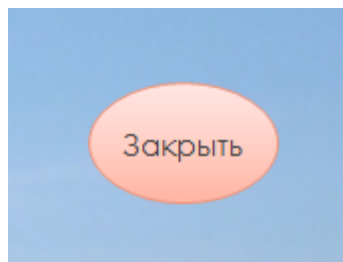


Рисунок 10 – Функциональная кнопка закрывающая базу данных сотрудников

ЗАКЛЮЧЕНИЕ

Таким образом, была разработана база данных, содержащая информацию о сотрудниках предприятия ООО «Мостовик-Восток». База данных разрабатывалась для отдела кадров, ее предназначение – обеспечить удобное добавление, хранение и поиск информации. Для разработки базы данных выбрана среда программирования Microsoft Access 2013, в ней были осуществлены все основные аспекты современных баз данных.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Бойко В.В., Савинков В.М. Проектирование баз данных информационных систем. – 2-е изд. – М.: Финансы и статистика, 2012. – 350 с.
2. Вольфенгаген В. Э., Кузин Л. Т., Саркисян В. И. Реляционные методы проектирования баз данных. – Киев: Вища Школа, 2012. – 192 с.
3. Вирт. Н., Алгоритмы и структуры баз данных / Н. Вирт – Москва : Мир, 2010. – 196 с.
4. Гурвиц Г.А., Microsoft Access 2010. Разработка приложений на реальном примере. СПб.: БХВ-Петербург, 2012. – 496 с.: ил. + CD-ROM – (Профессиональное программирование);
5. Горбатов В.А., Павлов П.Г., Четвериков В.Н. Логическое управление информационными процессами. – М.: Энергоиздат, 2014. – 304 с.
6. Голосов А.О., Цаленко М.Ш. Схемы реляционных баз данных: теория нормализации и построение нормальных форм // Прикладная информатика, вып.2. - М.: Финансы и статистика, 2012. – с. 92-119.
7. Грабер М., Введение в SQL. – М.: Лори, 2010.
8. Грей П., Логика, алгебра и базы данных. – М.: Машиностроение, 1989. – 360 с.
9. Дейт К.. Введение в системы баз данных. – М.: Вильямс, 2001. – 354 с.
10. Дейт. К., Введение в системы баз данных. – Киев-Москва: Диалектика, 2012. – 781 с.
11. Дрибас В.П., Реляционные модели баз данных. – Минск: БГУ, БССР, 2012. – 297 с.
12. Джексон Г., Проектирование реляционных баз данных для использования с микроЭВМ. – М.: Мир, 2011. – 252 с.
13. Йордан Э., Аргила К. Структурные модели в объектно-ориентированном анализе и проектировании. – М.: ЛОРИ, 2014. – 264 с.

14. Карпова И.П., Базы данных: Учебное пособие по курсу "Базы данных". – М., РИО МГИЭМ, 2011. – 118 с
15. Коннолли Т., Бегг К., Страчан А. Базы данных: проектирование, реализация, сопровождение. Теория и практика, 2-е изд. : Пер. с англ. : Уч. пос. – М.: Изд. дом "Вильямс", 2013. – 1120 с.
16. Копейкин М.В., Спиридонов В.В., Шумова Е.О. Базы данных. Основы SQL реляционных баз данных: Учеб.пособие. – СПб.: СЗТУ, 2014. – 177 с.
17. Тиори Т., Фрай Дж., Проектирование структур баз данных : В 2-х кн.: Пер. с англ. – М.: Мир, 2013.
18. Хоменко, А.Д., Базы данных. Учебник для ВУЗОВ / А.Д. Хоменко – Москва : Технология, 2011. – 325 с.

ПРИЛОЖЕНИЕ А

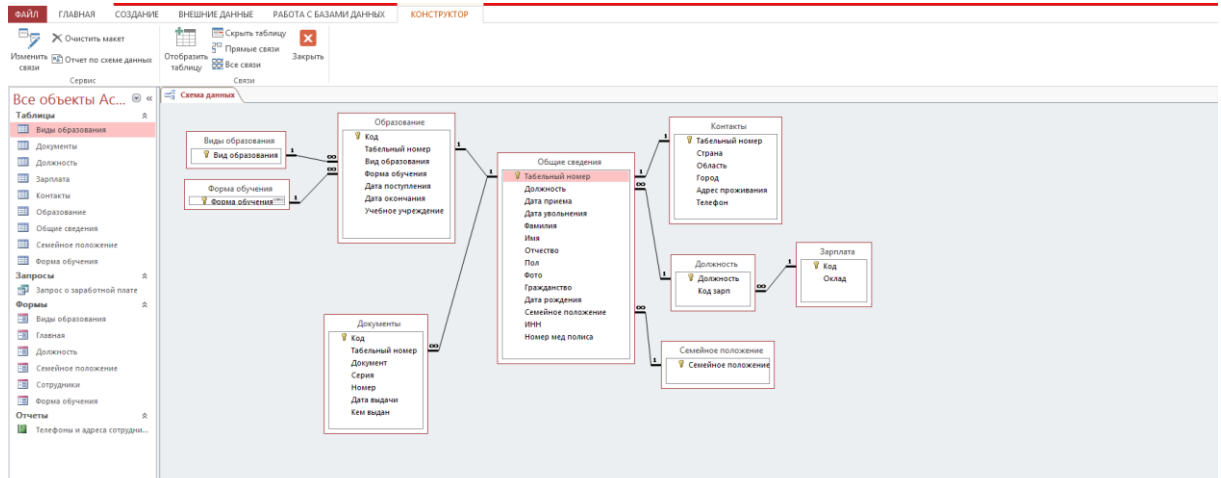


Рисунок 1 –Схема данных

The screenshot shows the Microsoft Access interface with the 'Общие сведения' table open. The table contains the following data:

Табельный номер	Должность	Дата приема	Дата уволи	Фамилия	Имя	Отчество	Пол	Фото	Гражданств	Дата рож	Семейное пол	ИНН	Номе	Щелкните для добавления
11201	Бригадир	09.03.2003		Вейде	Федор	Федорович	м		гражданин РФ	18.05.1970	женат			
11202	Водитель	01.08.2010		Воробьев	Сергей	Константинович	м		гражданин РФ	14.07.1976	женат			
11203	Крановщик	20.04.2010		Гес	Павел	Петрович	м		гражданин РФ	03.09.1980	женат			
11204	Бригадир	22.08.2005		Захаров	Павел	Сергеевич	м		гражданин РФ	14.09.1974	женат			
11205	Заместитель директора	24.04.2000		Ковалев	Станислав	Викторович	м		гражданин РФ	19.08.1964	женат			
11206	Уборщица производственных поме	18.03.2006	30.05.2013	Ковалева	Елена	Николаевна	ж		гражданин РФ	05.02.1966	замужем			
11207	Водитель	12.09.2010		Латеев	Игорь	Игоревич	м		гражданин РФ	15.02.1977	женат			
11208	Директор	04.04.2000		Мальчиков	Сергей	Викторович	м		гражданин РФ	07.08.1960	женат			
11209	Крановщик	30.05.2010		Матросов	Илья	Иванович	м		гражданин РФ	21.01.1983	женат			
11210	Сторож	07.04.2000	01.09.2007	Мирских	Сергей	Петрович	м		гражданин РФ	28.02.1960	женат			
11211	Главный бухгалтер	25.04.2000		Москвитин	Павел	Андреевич	м		гражданин РФ	27.10.1963	женат			
11212	Бухгалтер	08.02.2008		Нечаева	Татьяна	Владимировна	ж		гражданин РФ	17.04.1975	замужем			
11213	Водитель	02.03.2013		Петров	Алексей	Александрович	м		гражданин РФ	20.01.1980	женат			
11214	Сварщик	06.06.2014		Перевалов	Андрей	Сергеевич	м		гражданин РФ	10.06.1986	не женат			
11215	Сварщик	24.10.2004		Полосинский	Юрий	Александрович	м		гражданин РФ	27.09.1957	женат			
11216	Водитель	07.03.2002		Пономарев	Анатолий	Михайлович	м		гражданин РФ	10.11.1955	женат			
11217	Бетонщик	29.03.2005		Робатов	Сергей	Юревич	м		гражданин РФ	25.09.1969	женат			
11218	Эксоваторщик	13.06.2004		Родных	Юрий	Павлович	м		гражданин РФ	09.07.1970	женат			
11219	Эксоваторщик	14.03.2005		Соловьева	Дмитрий	Владимирович	м		гражданин РФ	30.08.1976	женат			
11220	Завхоз	05.09.2008		Соколов	Сергей	Иванович	м		гражданин РФ	16.11.1968	женат			
11221	Бетонщик	20.07.2005		Тарбеев	Николай	Викторович	м		гражданин РФ	05.11.1970	женат			
11222	Монтажник	04.07.2009		Тимафеев	Кирилл	Сергеевич	м		гражданин РФ	24.11.1979	женат			
11223	Сторож	29.09.2006		Фролов	Александр	Александрович	м		гражданин РФ	10.02.1978	женат			
11224	Инженер по технической безопаск	16.05.2005		Храмцова	Татьяна	Викторовна	ж		гражданин РФ	07.08.1960	не замужем			
11225	Монтажник	24.08.2007		Чумаченко	Николай	Михайлович	м		гражданин РФ	15.12.1968	женат			
11226	Бетонщик	14.04.2007	21.12.2009	Черкасн	Дмитрий	Николаевич	м		гражданин РФ	25.08.1980	женат			
11227	Монтажник	07.03.2013		Цыпкин	Сергей	Анатолеевич	м		гражданин РФ	22.01.1985	не женат			
11228	Сторож	01.02.2014		Шугин	Николай	Николаевич	м		гражданин РФ	30.09.1976	женат			
11229	Крановщик	01.03.2005	01.12.2009	Шумерски	Андрей	Федорович	м		гражданин РФ	30.03.1973	женат			
11230	Уборщица производственных поме	25.04.2000	01.12.2005	Шашина	Людмила	Михайловна	ж		гражданин РФ	01.05.1975	замужем			
11231	Бетонщик	01.06.2014		Шилов	Денис	Петрович	м		гражданин РФ	20.07.1982	не женат			
11232	Сварщик	11.07.2006	01.02.2012	Щевцов	Юрий	Сергеевич	м		гражданин РФ	31.07.1976	женат			

Риснок 2 – Таблица «Общие сведения»

Табельный номер	Область	Страна	Город	Адрес проживания	Телефон	Щелкните для добавления
11201	Красноярский край	Россия	Канск	ул.Шоссейная 12 кв. 35	251123	
11202	Красноярский край	Россия	Канск	ул.Иланская 30/2	556220	
11203	Красноярский край	Россия	Канск	ул.Кирова 12/1	221458	
11204	Красноярский край	Россия	Канск	ул.Пролетарская 7 кв. 77	854210	
11205	Красноярский край	Россия	Канск	ул.Пролетарская 10 кв. 10	114523	
11206	Красноярский край	Россия	Канск	ул.Московская 35 кв. 57	569632	
11207	Красноярский край	Россия	Канск	ул.Московская 12 кв. 15	221052	
11208	Красноярский край	Россия	Канск	ул.Шоссейная 20/2	114523	
11209	Красноярский край	Россия	Канск	ул.40 лет Октября 50 кв. 26	451255	
11210	Красноярский край	Россия	Канск	ул.Высшая 34	441202	
11211	Красноярский край	Россия	Канск	ул.40 лет Октября 10 кв. 76	456232	
11212	Красноярский край	Россия	Канск	ул.Севостопольская 56	221584	
11213	Красноярский край	Россия	Канск	ул.Краснопартизанская 30 кв. 89	112532	
11214	Красноярский край	Россия	Канск	ул.Шоссейная 32	452133	
11215	Красноярский край	Россия	Канск	ул.Рабочая 53 кв.17	456521	
11216	Красноярский край	Россия	Канск	м-н Солнечный ул.Береговая 32 кв. 50	223654	
11217	Красноярский край	Россия	Канск	ул.Краснопартизанская 22 кв. 20	112563	
11218	Красноярский край	Россия	Канск	ул.Циллиная 45	545632	
11219	Красноярский край	Россия	Канск	ул.Московская 24 кв. 43	414520	
11220	Красноярский край	Россия	Канск	ул.Иланская 12/1	456522	
11221	Красноярский край	Россия	Канск	ул.Береговая 25 кв. 52	774125	
11222	Красноярский край	Россия	Канск	ул.Калинина 12 кв. 34	445210	
11223	Красноярский край	Россия	Канск	ул.Паринской Коммуны 67 кв. 55	456521	
11224	Красноярский край	Россия	Канск	ул.Кайтымская 33 кв.45	778562	
11225	Красноярский край	Россия	Канск	ул.Костерная 12 кв.34	125632	
11226	Красноярский край	Россия	Канск	ул.Пролетарская 10 кв. 99	452123	
11227	Красноярский край	Россия	Канск	м-н Солнечный ул. Белых 33 кв. 50	256321	
11228	Красноярский край	Россия	Канск	ул.Рабочая 40 кв. 78	125482	
11229	Красноярский край	Россия	Канск	ул.Дзержинского 60 кв. 17	523552	
11230	Красноярский край	Россия	Канск	ул.Московская 2 кв. 5	452103	
11231	Красноярский край	Россия	Канск			
11232	Красноярский край	Россия	Канск			

Рисунок 3 – Таблица «Контакты»

Код	Табельный	Документ	Серия	Номер	Дата выдачи	Кем выдан	Щелкните для добавления
1	11201	паспорт	0425			УВМС	
2	11202	паспорт	0463			УВМС	
3	11203	паспорт	0442			УВМС	
4	11204	паспорт	0485			УВМС	
5	11205	паспорт	0421			УВМС	
6	11206	паспорт	0496			УВМС	
7	11207	паспорт	0436			УВМС	
8	11208	паспорт	0412			УВМС	
9	11209	паспорт	0454			УВМС	
10	11210	паспорт	0441			УВМС	
11	11211	паспорт	0448			УВМС	
12	11212	паспорт	0488			УВМС	
13	11213	паспорт	0432			УВМС	
14	11214	паспорт	0499			УВМС	
15	11215	паспорт	0496			УВМС	
16	11216	паспорт	0475			УВМС	
17	11217	паспорт	0477			УВМС	
18	11218	паспорт	0473			УВМС	
19	11219	паспорт	0476			УВМС	
20	11220	паспорт	0455			УВМС	
21	11221	паспорт	0433			УВМС	
22	11222	паспорт	0485			УВМС	
23	11223	паспорт	0450			УВМС	
24	11224	паспорт	0459			УВМС	
25	11225	паспорт	0410			УВМС	
26	11226	паспорт	0421			УВМС	
27	11227	паспорт	0459			УВМС	
28	11228	паспорт	0411			УВМС	
29	11229	паспорт	0415			УВМС	
30	11230	паспорт	0485			УВМС	
31	11231	паспорт	0447			УВМС	
32	11232	паспорт	0466			УВМС	
(№)							

Рисунок 4 – Таблица «Документы»

РАБОТА С ТАБЛИЦАМИ | Сотрудники : база данных - C:\Users\...\Desktop\Сотрудники.accdb (Формат файлов Access 2007-2013) - Access

РАБОТА С БАЗАМИ ДАННЫХ | ТАБЛИЦА

ВНЕШНИЕ ДАННЫЕ | РАБОТА С БАЗАМИ ДАННЫХ | ПОЛЯ | ТАБЛИЦА

Вырезать | Копировать | Вставить | Буфер обмена | По возрастанию | По убыванию | Удалить сортировку | Фильтр | Выделение | Дополнительно | Создать | Итоги | Сохранить | Орфография | Заменить | Найти | Перейти | Выбрать | Ж К Н | Форматирование текста

Режимы | Буфер обмена | Сортировка и фильтр | Записи

Все объекты Ас... | Главная | Сотрудники | Общие сведения | Контакты | Документы | Образование

Код	Табельный	Вид образования	Форма обуче	Дата поступления	Дата оконч	Учебное учрежде	Щелкните для добавления
1	11201	Средне техническое	очная			ПТУ	
2	11202	Средне специальное	очная			ПТУ	
3	11203	Средне специальное	очная			ПТУ	
4	11204	Средне техническое	заочная			ПТУ	
5	11205	Высшее	очная			ПТУ	
6	11206	Средне специальное	заочная			ПТУ	
7	11207	Средне специальное	очная			ПТУ	
8	11208	Высшее	очная			ПТУ	
9	11209	Средне техническое	очная			ПТУ	
10	11210	Средне специальное	заочная			ПТУ	
11	11211	Высшее	очная			ПТУ	
12	11212	Высшее	очная			ПТУ	
13	11213	Средне специальное	заочная			ПТУ	
14	11214	Средне техническое	очная			ПТУ	
15	11215	Средне техническое	очная			ПТУ	
16	11216	Средне специальное	очная			ПТУ	
17	11217	Средне техническое	очная			ПТУ	
18	11218	Средне техническое	очная			ПТУ	
19	11219	Средне техническое	очная			ПТУ	
20	11220	Высшее	заочная			ПТУ	
21	11221	Средне техническое	очная			ПТУ	
22	11222	Средне специальное	очная			ПТУ	
23	11223	Средне специальное	очная			ПТУ	
24	11224	Высшее	очная			ПТУ	
25	11225	Средне техническое	очная			ПТУ	
26	11226	Средне специальное	очная			ПТУ	
27	11227	Средне специальное	очная			ПТУ	
28	11228	Средне специальное	очная			ПТУ	
29	11229	Средне специальное	очная			ПТУ	
30	11230	Средне техническое	заочная			ПТУ	
31	11231	Средне специальное	очная			ПТУ	
32	11232	Средне специальное	очная			ПТУ	
(№)							

Рисунок 5 – Таблица «Образование»