



Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Магнитогорский государственный технический университет им. Г.И. Носова»

М.В. Махмутова

УПРАВЛЕНИЕ ДАННЫМИ

*Утверждено Редакционно-издательским советом университета
в качестве учебно-методического пособия*

Магнитогорск
2021

УДК 657.1.011.56
ББК 32.972.13

Рецензенты:

директор ИТ-департамента компании ООО «Технологии плюс»
И.В. Чумичев

кандидат педагогических наук,
доцент кафедры прикладной математики и информатики,
ФГБОУ ВО «Магнитогорский государственный технический
университет им. Г.И. Носова»
Е.А. Москвина

Махмутова М.В.

Управление данными [Электронный ресурс] : учебно-методическое пособие / Марина Владимировна Махмутова ; ФГБОУ ВО «Магнитогорский государственный технический университет им. Г.И. Носова». – Электрон. текстовые дан. (1,40 Мб). – Магнитогорск : ФГБОУ ВО «МГТУ им. Г.И. Носова», 2021. – 1 электрон. опт. диск (CD-R). – Систем. требования : IBM PC, любой, более 1 GHz ; 512 Мб RAM ; 10 Мб HDD ; MS Windows XP и выше ; Adobe Reader 8.0 и выше ; CD/DVD-ROM дисковод ; мышь. – Загл. с титул. экрана.

ISBN 978-5-9967-2099-6

Учебно-методическое пособие «Управление данными» предлагается для студентов следующих направлений подготовки: 09.04.03 «Прикладная информатика», 38.03.05 «Бизнес-информатика», разработано в поддержку курса Управление данными. В учебно-методическом пособии рассматриваются вопросы по основным теоретическим положениям технологии управления данными, даны рекомендации по выполнению лабораторных работ и для самостоятельной подготовки обучающихся, выполнение которых направлено на формирование важных профессиональных компетенций.

Материалы пособия могут быть использованы при изучении дисциплин «Управление данными», «Базы данных», «Хранилища данных», «Предметно-ориентированные экономические информационные системы» очной и заочной форм обучения.

УДК 657.1.011.56
ББК 32.972.13

ISBN 978-5-9967-2099-6

© Махмутова М.В., 2021

© ФГБОУ ВО «Магнитогорский государственный
технический университет им. Г.И. Носова», 2021

Содержание

ВВЕДЕНИЕ	4
1. ЦЕЛИ И ЗАДАЧИ ДИСЦИПЛИНЫ	7
2. ОСНОВНЫЕ ПОНЯТИЯ	11
3. ОСНОВНЫЕ РАЗДЕЛЫ ДИСЦИПЛИНЫ И ИХ СОДЕРЖАНИЕ	13
3.1. Банки данных.....	13
3.2. Архитектура системы баз данных.....	18
3.3. Защита данных в системах с базами данных	26
3.4. Управление системами хранения данных	31
4. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ И ПОДГОТОВКА К МЕРОПРИЯТИЯМ ТЕКУЩЕГО КОНТРОЛЯ	39
5. МАТЕРИАЛЫ ПО КОНТРОЛЮ ЗНАНИЙ	41
6. ЗАДАНИЯ ДЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ	43
ГЛОССАРИЙ	49
БИБЛИОГРАФИЧЕСКИЙ СПИСОК.....	51
ПРИЛОЖЕНИЯ	53

ВВЕДЕНИЕ

В учебно-методическом пособии рассмотрены основные понятия технологии управления данными. Представлены основные практические подходы, средства автоматизации процесса проектирования. Рассмотрены этапы процесса проектирования базы данных, проблемы автоматизации каждого этапа.

По мере развития информационных технологий совершенствовались методы решения экономических задач на компьютерах. На начальном этапе задачи решались изолированно, а прикладные программы, реализующие их, сами обеспечивали ввод и организацию необходимых данных. Такая технология приводила к значительному дублированию хранимых данных и затрудняла их обновление. Большие объемы экономической информации, ее относительно высокая стабильность, наряду с требованиями к актуальности и достоверности, привели к необходимости интеграции данных в единой базе, обеспечивающей решение всего комплекса задач управления предприятием. В этой связи потребовалось разработать специальные методы и механизмы управления такого рода совместно используемыми ресурсами данных, которые получили название хранилищ данных. На концепции хранилищ данных базируются основные идеи современной информационной технологии администрирования и управления данными. Увеличение объема и структурной сложности хранимых данных, расширение круга пользователей информационных систем выдвинуло требования создания удобных общесистемных средств интеграции хранимых данных и управления ими.

Хранилища данных (Data Warehouse) представляют собой специализированные базы данных, предназначенные для хранения данных, которые редко меняются, но, на основе которых часто требуется выполнение сложных запросов. Обычно они ориентированы на выполнение аналитических запросов, которые обеспечивают поддержку принятия решений для руководителей и менеджеров. Хранилища данных позволяют разгрузить промышленные базы данных, и тем самым позволяют пользователям более эффективно и быстро извлекать необходимую информацию.

Как правило, хранилища данных оперируют с огромными объемами информации, что предъявляет к их проектированию и реализации повышенные требования. Выбор в качестве платформы хранилища данных такой высокопроизводительной РСУБД позволяет существенно повысить общую эффективность создаваемой информационной системы. В этом случае ERwin (CASE-средство фирмы PLATINUM Technology inc.) становится незаменимым инструментом, поскольку с одной стороны эффективно поддерживает на физическом уровне проектирование объектов РСУБД, с другой стороны имеет специализированные средства моделирования хранилищ данных.

К проектированию хранилищ данных обычно предъявляются следующие требования:

- 1) структура данных хранилища должна быть понятна пользователям;

2) должны быть выделены статические данные, которые регулярно модифицируются: ежедневно, еженедельно, ежеквартально;

3) должны быть упрощены требования к запросам, с целью исключения запросов, которые могли бы требовать множественных утверждений SQL в традиционных реляционных СУБД;

4) должна быть обеспечена поддержка сложных запросов SQL, которые требуют последовательной обработки тысяч или миллионов записей.

Эти требования существенно отличают структуру реляционных СУБД и хранилищ данных. Нормализация данных в реляционных СУБД приводит к созданию множества связанных между собой таблиц. В результате, выполнение сложных запросов неизбежно приводит к объединению многих таблиц, что существенно увеличивает время отклика. Проектирование хранилища данных подразумевает создание денормализованной структуры данных (допускается избыточность данных и возможность возникновения аномалий при манипулировании данными), ориентированной в первую очередь на высокую производительность при выполнении аналитических запросов. Нормализация делает модель хранилища слишком сложной, затрудняет ее понимание и ухудшает эффективность выполнения запроса.

Для эффективного проектирования хранилищ данных используется многомерная или размерная (Dimensional) модель. Dimensional - методология проектирования, специально предназначенная для разработки хранилищ данных. Моделирование Dimensional сходно с моделированием связей и сущностей для реляционной модели, но отличается целями. Реляционная модель делает упор на целостность и эффективность ввода данных. Размерная (Dimensional) модель ориентирована в первую очередь на выполнение сложных запросов к БД.

Учебно-методическое пособие ориентировано на студентов очного и заочного отделений, изучающих технологии управления данными, системами с базами данных, методологии моделирования данных, проблемы проектирования систем с базами данных, проблемы стандартизации в области технологий организации и управления данными. Лабораторные работы в учебно-методическом пособии посвящены проблемам собственно разработки модели данных хранилища данных. Определим основные понятия дисциплины.

В размерном моделировании принят стандарт модели, называемый схемой звезда (star schema), которая обеспечивает высокую скорость выполнения запроса посредством денормализации и разделения данных. Невозможно создать универсальную денормализованную структуру данных, обеспечивающую высокую производительность при выполнении любого аналитического запроса. Поэтому схема звезда строится так, чтобы обеспечить наивысшую производительность при выполнении одного самого важного запроса, либо для группы похожих запросов.

Схема звезда обычно содержит одну большую таблицу, называемую таблицей факта (fact table), помещенную в центр, и окружающие ее меньшие таблицы, называемые таблицами размерности (dimensional table), соединенными с таблицей факта в виде звезды радиальными связями. В этих

связях таблицы размерности являются родительскими, таблица факта-дочерней. Схема звезда может иметь также консольные таблицы (outrigger table), присоединенные к таблице размерности. Консольные таблицы являются родительскими, таблицы размерности - дочерними.

Прежде чем создать базу данных со схемой типа звезда, необходимо проанализировать бизнес-правила предметной области, с целью выяснения центрального вопроса, ответ на который наиболее важен. Все прочие вопросы должны быть объединены вокруг этого основного вопроса и моделирование должно начинаться с этого основного вопроса. Данные, необходимые для ответа на этот вопрос, должны быть помещены в центральную таблицу модели - таблицу факта.

Главной задачей проектирования базы данных для хранилища является определение состава и структуры базы данных, способа ее организации, выбор инструментальных средств управления данными.

Учебно-методическое пособие «Управление данными» представлено: введением, четырьмя разделами с теоретическим материалом и практическими заданиями, разделом с материалами по контролю знаний, которые включают контрольную работу и вопросы для подготовки к зачету, раздел с заданиями для самостоятельной работы студентов, глоссарий и список необходимых источников для ознакомления и изучения, семь приложений с примерами выполнения заданий лабораторных работ.

Об авторе: Махмутова Марина Владимировна, доцент, канд. пед. наук, доцент кафедры бизнес-информатики и информационных технологий института энергетики и автоматизированных систем ФГБОУ ВО «Магнитогорский государственный технический университет им. Г.И.Носова». Является ведущим преподавателем по дисциплинам «Управление данными», «Технологии баз данных и СУБД», «Хранилища данных», «Предметно-ориентированные экономические информационные системы», «Проектирование ИС». Основная сфера научных интересов – информационные системы, базы данных, проектирование, моделирование, жизненный цикл автоматизированных информационных систем, информационно-образовательная среда подготовки специалистов в сфере информационных технологий. Является автором более 50 публикаций.

1. ЦЕЛИ И ЗАДАЧИ ДИСЦИПЛИНЫ

Содержание учебно-методического пособия отвечает требованиям ФГОС ВО, соответствует учебному плану по направлению подготовки и рабочей программе дисциплины «Управление данными».

Целью дисциплины «Управление данными» является повышение исходного уровня владения современными технологиями управления данными, представленными в виде баз данных, достигнутого на предыдущей ступени образования, для решения профессиональных задач в области разработки прикладных решений для бизнеса.

Для достижения поставленной цели в курсе «Управление данными» решаются задачи:

1) формирование более глубоких знаний

–о современных технологиях организации, хранения, преобразования и использования систем с базами данных и системах управления базами данных (СУБД),

–о математических основах теории баз данных и управления данными,

–о методах и средствах проектирования систем с базами данных и работы с ними;

2)отработка умения и навыков информационного моделирования, проектирования и эксплуатации систем с базами данных.

В ходе выполнения заданий лабораторного практикума, студенты познакомятся с современными программными средствами: MicrosoftAccess, MY SQL, Microsoft SQL Server и др.

Целью разработки учебно-методического пособия «Управление данными» является обеспечение преподавателей и студентов теоретическим, практическим и методическим материалами для подготовки по курсу «Управление данными» в соответствии с требованиями ФГОС ВО по направлениям подготовки.

В рамках дисциплины предполагается решение следующих задач:

1) изучение теоретических вопросов по созданию, структуре, принципам функционирования современных автоматизированных информационных систем с базами данных;

2) знакомство с концепцией построения автоматизированных информационных систем, предназначенных для решения оперативных функциональных задач, задач аналитической обработки и интеллектуального анализа в системах поддержки принятия решений;

3) проведение анализа основных систем с базами данных и обоснование выбора СУБД для реализации базы данных для решения проблем объектов автоматизации;

4) получение практических навыков по созданию баз данных в среде различных СУБД;

5) получение представления о системном исследовании экономических объектов с целью их декомпозиции на подсистемы и определению на основе

целевых критериев состава, содержания и взаимосвязи экономических задач в условиях их автоматизации.

Дисциплина «Управление данными» входит в базовую часть учебного плана образовательной программы. Особенности изучения дисциплины:

а) для изучения данной дисциплины студенты должны иметь теоретические и практические знания по дисциплинам: «Информационные системы и технологии», «Методологии и технологии моделирования бизнес-процессов», «Базы данных», «Интернет-технологии», «Теория вероятностей и математическая статистика для анализа больших данных», «Учебная - ознакомительная практика». Данная дисциплина базируется на знаниях цикла дисциплин: «Учебная - практика по получению первичных профессиональных умений и навыков, в том числе первичных умений и навыков научно-исследовательской деятельности», «Информационные системы управления взаимоотношениями с клиентами», «Основы научно-исследовательской работы в сфере ИКТ» и др.;

б) при изучении каждой темы студент должен придерживаться следующего порядка: изучить теоретический материал, ответить на вопросы для самоконтроля, выполнить задания для самостоятельной работы, выполнить лабораторную работу, элементы индивидуального задания;

с) переход к изучению новой темы возможен в случае выполнения всех заданий предыдущих разделов.

Предлагаются следующие требования к уровню освоения содержания курса, представлены в таблице 1.

В результате изучения дисциплины студент должен знать:

1) инструменты и методы моделирования бизнес-процессов анализа и управления данными в ИС и обменом с другими системами;

2) современные методы и инструментальные средства прикладной информатики; инструменты и методы функционального и информационного моделирования;

3) основные понятия технологии управления обменом данными в ИС и с другими участниками процесса; протоколы и форматы обмена данными.

Студент должен овладеть следующими умениями и навыками:

1) анализировать исходную документацию по обмену информацией в организации;

2) анализировать функциональные возможности технологии обмена данными в ИС;

3) моделировать бизнес-процессы с учетом их последующей адаптации;

4) разрабатывать проектные решения по управлению интеграцией данных;

5) решать типовые прикладные задачи на основе технологии обмена данными в ИС с использованием средств автоматизации

6) навыками разработки проектных решений по управлению интеграцией данных у заказчика;

7) навыками построения модели управления обменом информации в ИС;

Таблица 1

Дескрипторы индикаторов достижения сформированности компетенций

Структурный элемент компетенции	Планируемые результаты обучения
ПК-3 Способен применять современные методы и инструментальные средства прикладной информатики для автоматизации и информатизации решения прикладных задач различных классов	
ПК-3.1 Моделирует бизнес-процессы, выполняет работы по их адаптации (реверс-инжинирингу)	
Знать	Инструменты и методы моделирования бизнес-процессов анализа и управления данными в ИС и обменом с другими системами
Уметь	Анализировать исходную документацию по обмену информацией в организации. Анализировать функциональные возможности технологии обмена данными в ИС. Моделировать бизнес-процессы с учетом их последующей адаптации.
Владеть	Навыками построения модели управления обменом информации в ИС
ПК-3.2 Осуществляет управление требованиями в ИТ-проектах для автоматизации и информатизации решения прикладных задач различных классов	
Знать	Основные понятия технологии управления обменом данными в ИС и с другими участниками процесса; протоколы и форматы обмена данными
Уметь	Решает типовые прикладные задачи на основе технологии обмена данными в ИС с использованием средств автоматизации
Владеть	Навыками принятия управленческого решения по автоматизации выполнения прикладных задач управления обменом информации в ИС и с другими участниками процесса произвольного класса сложности
ПК-3.3 Осуществляет управление и разработку проектных решений по автоматизации и информатизации профессиональных задач	
Знать	Современные методы и инструментальные средства прикладной информатики; инструменты и методы функционального и информационного моделирования
Уметь	Разрабатывать проектные решения по управлению интеграцией данных.
Владеть	Навыками разработки проектных решений по управлению интеграцией данных у заказчика. Навыками формирования пользовательской документации к информационной базе с учетом проектных решений и принятых в организации или проекте стандартов и технологий

8) навыками принятия управленческого решения по автоматизации выполнения прикладных задач управления обменом информации в ИС и с другими участниками процесса произвольного класса сложности;

9) навыками формирования пользовательской документации к информационной базе с учетом проектных решений и принятых в организации или проекте стандартов и технологий

Студент должен иметь представление:

1) о перспективах развития информационных технологий и информационных систем в бизнесе;

2) об этических и социальных последствиях внедрения информационных технологий.

В ходе теоретической подготовки предусматривается:

– использование электронного демонстрационного материала в форме презентаций;

– использование электронного демонстрационного материала по темам, требующим иллюстрации работы программных продуктов;

– организация дискуссии по теме «Проблема эффективности использования ресурсов информационных систем».

В ходе проведения всех практических занятий предусматривается использование средств вычислительной техники и ИКТ при выполнении заданий и подготовки к семинарам и контрольным работам.

В ходе изучения дисциплины рекомендуется использовать образовательные технологии:

– возможности образовательного портала ФГБОУ ВО «МГТУ им. Г.И. Носова» для предоставления студентам графика самостоятельной работы, расписания консультаций, заданий для самостоятельного выполнения и рекомендуемых тем для самостоятельного изучения;

– традиционные технологии обучения в виде занятий с использованием мультимедийных средств и лабораторных практикумов в компьютерных классах вычислительного центра ФГБОУ ВО «МГТУ им. Г.И. Носова».

2. ОСНОВНЫЕ ПОНЯТИЯ

Администратор базы данных - лицо, отвечающее за выработку требований к базе данных, ее проектирование, реализацию, эффективное использование и сопровождение.

База данных (БД)- совокупность связанных данных, организованных по определенным правилам, предусматривающим общие принципы описания, хранения и манипулирования, независимая от прикладных программ. База данных является информационной моделью предметной области. Обращение к базам данных осуществляется с помощью системы управления базами данных (СУБД). По законодательству РФ база данных - объективная форма представления и организации совокупности данных, систематизированных таким образом, чтобы эти данные могли быть найдены и обработаны с помощью ЭВМ.

Банк данных (БнД) - это система специальным образом организованных данных – баз данных, программных, технических, языковых, организационно-методических средств, предназначенных для обеспечения централизованного накопления и коллективного многоцелевого использования данных. В состав банка данных входят одна или несколько баз данных, справочник баз данных, СУБД, а также библиотеки запросов и прикладных программ.

Государственная база данных - в РФ база данных, созданная, приобретенная или накапливаемая за счет или с привлечением средств федерального бюджета.

Данные - это любая информация об объектах окружающего мира, представленная в формализованном виде, пригодном для ее передачи, хранения и обработки при помощи некоторого процесса с использованием средств вычислительной техники.

Документографическая база данных - база данных, запись в которой отражает конкретный документ, содержит его библиографическое описание и, возможно, иную информацию о нем.

Локальная база данных - база данных, размещенная на одном или нескольких носителях на одном компьютере.

Основные функции группы администратора БД: анализ предметной области; проектирование структуры БД; задание ограничений целостности при описании структуры БД и процедур обработки БД; первоначальная загрузка и ведение БД; защита данных; обеспечение восстановления БД; анализ обращений пользователей БД; анализ эффективности функционирования БД; работа с конечными пользователями; подготовка и поддержание системных средств.

Объектно-ориентированная база данных - база данных, в которой данные оформлены в виде моделей объектов, включающих прикладные программы, которые управляются внешними событиями.

Ограничения целостности - условия, которым должны удовлетворять хранимые в базе данные.

Отсылочная база данных - база данных, отсылающая пользователя к другим источникам для получения полной или дополнительной информации.

Портативная база данных - база данных, предоставляемая пользователям непосредственно, минуя коммуникационную сеть.

Предметная область - это часть реального мира, подлежащего изучению для организации управления и, в конечном счете, автоматизации.

Распределенная база данных - территориально распределенная совокупность локальных баз данных, объединенных согласованными принципами организации, комплектования и эксплуатации, а также каналами связи, и доступная для совместного использования.

Реляционная база данных - база данных, построенная на основе реляционной модели. В реляционной базе каждый объект задается записью (строкой) в таблице. Реляционная база создается и затем управляется с помощью реляционной системы управления базами данных.

Система управления базами данных - комплекс программных и лингвистических средств общего или специального назначения, реализующий поддержку создания баз данных, централизованного управления и организации доступа к ним различных пользователей в условиях принятой технологии обработки данных. СУБД характеризуется используемой моделью, средствами администрирования и разработки прикладных процессов. СУБД обеспечивает: описание и сжатие данных; манипулирование данными; физическое размещение и сортировку записей; защиту от сбоев, поддержку целостности данных и их восстановление; работу с транзакциями и файлами; безопасность данных. СУБД определяет модель представления данных.

Структура базы данных - принцип или порядок организации записей в базе данных и связей между ними.

Текстовая база данных - база данных, записи в которой содержат (главным образом) текст на естественном языке.

Тиражирование данных - в распределенных базах данных - технология распространений изменений, первоначально выполненных на одной копии блока данных, на другие копии. Различают: синхронное тиражирование, предусматривающее практически одновременное изменение данных во всех частях базы; асинхронное тиражирование, предусматривающее изменение данных по очереди.

Фактографическая база данных - база данных, содержащая информацию, относящуюся непосредственно к предметной области.

Целостность базы данных - свойство базы данных, означающее, что БД содержит полную и непротиворечивую информацию, необходимую для корректного функционирования приложений. Для обеспечения целостности накладывают ограничения целостности.

Централизованная база данных - база данных, размещенная в виде единого информационного массива на одном или нескольких носителях в одной ЭВМ.

3. ОСНОВНЫЕ РАЗДЕЛЫ ДИСЦИПЛИНЫ И ИХ СОДЕРЖАНИЕ

3.1. Банки данных

3.1.1. Информация и данные

Рассматриваются основные понятия банков данных и знаний, предметная область банка данных, банк данных как автоматизированная система, архитектура банка данных, пользователи банков данных, администратор базы данных, тенденции развития банков данных.

Цель изучения: сформировать представление о предметной области и решаемых задачах базы данных, банка данных, хранилища данных, их архитектуре.

Теоретическая часть

Области использования вычислительной техники:

1. Выполнение численных расчетов;
2. Автоматические или автоматизированные информационные системы.

Информационная система (ИС) - это совокупность структурированных данных и комплекса программно-аппаратных средств для хранения данных и манипулирования ими.

Объект - это нечто существующее и различимое, то есть имеется способ отличить один подобный объект от другого.

Данное - это определенный показатель, который характеризует объект и принимает для конкретного экземпляра этого объекта некоторое значение.

Файл - это именованная область внешней памяти, в которую можно записывать и из которой можно считывать данные. С точки зрения пользователя файл содержит линейную последовательность записей. Стандартные операции: создать файл (требуемого типа и размера); открыть ранее созданный файл; прочитать из файла некоторую запись (текущую, следующую, предыдущую, первую, последнюю); записать в файл на место текущей записи новую, добавить новую запись в конец файла.

Файл последовательного доступа представляет собой последовательность записей данных в виде строк произвольной длины, разделенных запятыми или специальными символами, обозначающими переход на новую строку. Особенности: отсутствие возможности упорядочить хранимые записи; размещение и извлечение записей в такой файл производится построчно в определенной последовательности.

Файл произвольного доступа состоит из записей фиксированной длины, которая указывается при его создании. Особенности: все записи в упорядочены, каждая имеет свой номер; возможность быстрого перемещения на любую запись, минуя предыдущие.

Недостатки традиционного подхода: избыточность данных, проблемы непротиворечивости данных, ограниченная доступность данных, сложности в организации и управлении, недостаточность средств защиты хранимых данных,

низкопроизводительная работа в многопользовательской среде, отсутствие процедур восстановления данных после возникновения отказов, отсутствие средств манипулирования данными, высокая стоимость программирования и сопровождения, негибкость к изменениям и др..

Требования к современной СУБД: эффективное выполнение различных функций предметной области, минимизация и контроль избыточности хранимых данных, предоставление для принятия решений непротиворечивой (согласованной) информации, обеспечение возможности одновременного доступа к базе данных нескольких уполномоченных пользователей, обеспечение управления безопасностью, простая физическая реорганизация, т.е. изменение структуры данных в базе данных, возможность централизованного управления базой данных.

Для самоконтроля по теме необходимо ответить на следующие вопросы:

1. Дайте характеристику информационно-поисковым системам.
2. Дайте характеристику системам обработки данных.
3. Какие системы относятся к фактографическим ИС?
4. Что является результатом запроса в документальной ИС?
5. Основной принцип архитектуры БД ANSI.
6. Какие бывают системы баз данных по характеру использования?
7. Что такое распределенная система баз данных?
8. В чем суть отличия обработки информации в БД по принципу файл/сервер и клиент/сервер?
9. Какие операторы определены в языке определения данных?
10. Какие операторы определены в языке манипулирования данными?

Содержание лабораторных занятий.

ЛР 1. Анализ предметной области. Постановка задачи

ЛР 2. На основе результатов анализа предметной области (лабораторная работа 1) выделить группы пользователей, основные информационные потоки и функции обмена данными.

Пример выполнения заданий лабораторных работ представлен в приложении 1.

3.1.2. Базы данных

Рассматриваются следующие вопросы и понятия: база данных как информационная модели предметной области, система управления базой данных, основы теории реляционных баз данных, архитектура систем с базами данных, инфологическое проектирование базы данных, представление структур данных в памяти ЭВМ, проектирование баз данных, обзор промышленных СУБД, новые технологии в области баз данных.

Цель изучения: сформировать общее представление о системе стандартов баз данных.

Теоретическая часть

Система стандартов технологий баз данных (изучается самостоятельно, контроль в форме семинара).

Стандарты баз данных, их назначение и виды на примере международных стандартов ISO TC97/SC5/WG3 (Concepts and terminology for the conceptual schema and the information base), ISO/IEC 8907:1987, стандартов де-факто группы ANSI/X3/SPARC, системы стандартов по базам данных (ГОСТ 34.321-96 «Эталонная модель управления данными», ГОСТ 34.320-96 «Концепции и терминология для концептуальной схемы и информационной базы»).

С точки зрения решения конкретной задачи всю информацию о некоем объекте можно условно считать состоящей из трех взаимосвязанных пересекающихся, нечетко делимых частей: знания, протознания, информационный мусор.

Знания в отличие от сигналов и данных это уже элемент информации, отличающийся логической полнотой, ограниченностью набора, и, в завершённом варианте, проверенностью практикой. Знания включают в себя всю релевантную информацию, т.е. ту часть "полезной" для решения стоящей задачи информации, которая полностью устраняет (с точки зрения решаемой задачи!) неопределенность об объекте у получателя.

Релевантная информация, есть часть полезной (для получателя) информации - совокупности сведений, уменьшающих степень неопределенности о рассматриваемом объекте у ее получателя. В отличие от знаний полезная информация может быть логически не полной, противоречивой, не всегда достоверной и т.п. Знания же это полезная информация, представленная в виде, удобном для ее интеллектуального «переваривания».

Протознания - это та часть информации, из которой могут быть получены новые знания.

Остальная часть информации, - это информационный мусор - содержащиеся в «данных» сведения, которые не содержат полезной для пользователя информации и/или протознаний, но многократно увеличивают издержки пользователя.

Сбор данных - действия с «данными», не приводящие к изменению семантики.

Обработка данных (сигналов, данных, знаний) - действия, направленные на приведение интересующей исследователя, содержащейся в них информации к более удобному, компактному для анализа виду.

Заметим, что при обработке «данных» содержащееся в них количество информации не может быть увеличено. Здесь происходит лишь преобразование информации к виду, более удобному для дальнейших операций.

Под анализом данных будем понимать действия с ними, направленные на извлечение из них содержащейся информации об исследуемом объекте и на получение по имеющимся «данным» новых «данных», включающих в себя извлечённую из первых информацию об объекте.

Интеллектуальный анализ данных – это разновидность анализа данных, связанный, во-первых, именно с неформальным извлечением знаний об исследуемом объекте, породившем данные, непосредственно из этих данных (ИАД-1), во-вторых, с получением новых знаний об объекте на базе

извлечённых знаний, виртуальных данных об объекте и естественного и/или искусственного интеллекта (ИАД-2), в-третьих, с поиском, выбором, синтезом методов и средств обработки и анализа данных с учётом поставленных целей исследования объекта (ИАД-3).

Под применением результатов обработки и анализа данных понимаются действия, связанные с использованием результатов для решения теоретических и практических задач, с реализацией технологии, с достижением поставленной пользователем конечной цели. Заметим, что любая формализация, структурирование, обработка данных привносит в них элементы знаний.

Data Mining (DM) понимают как поиск характеризующих объект скрытых зависимостей и взаимосвязей, проявляющихся через данные о нем.

Data-Based Knowledge (DBK) - знания, основанные на данных, т.е. знания, которые «извлекаются» (выводятся) с помощью интеллектуальных инструментальных средств анализа (Business Intelligence Tools - BIT) из хранилища данных.

Data Cleaning (DC) очистка данных - есть устранение из данных или корректировка той их части, которая содержит легко определяемый информационный мусор: ошибки, противоречия, сбои, явные выбросы и т.п. Эта операция обычно используется перед загрузкой данных в хранилище (DW).

Data Warehouse (DW) - хранилище данных - это база, в которой собираются данные для последующего их анализа под конкретную решаемую задачу.

Metadata - данные, описывающие данные в DW.

Для развития и процветания бизнеса недостаточно просто владеть информацией. Все зависит от качества владения, т.е. умения четко и быстро определять категорию показателей, на основании которых требуется принимать решение; проводить анализ определенных значений данных показателей с различной степенью детализации с учетом всех параметров, оказывающих на них влияние.

Хранилище данных (Data Warehouse) по сути, представляет собой центр, в который собирается вся необходимая информация из различных подразделений предприятия. Прежде чем попасть в хранилище, данные должны быть соответствующим образом обработаны.

В основе концепции хранилища данных лежат две основные идеи: интеграция разьединенных детализированных данных (описывающих некоторые конкретные факты, свойства, события и т.д.) в едином хранилище и разделение наборов данных и приложений, используемых для обработки и анализа.

Определение понятия «хранилище данных» первым дал Уильям Инмон в своей монографии [W. Inmon, Building the Data Warehouse. John Willey & Sons, New York, 1992.] это «предметно-ориентированная, интегрированная, содержащая исторические данные, неразрушаемая совокупность данных, предназначенная для поддержки принятия управленческих решений».

Концептуально модель хранилища данных можно представить в виде схемы. Данные из различных источников помещаются в хранилище, а их

описания в репозиторий метаданных. Конечный пользователь, используя различные инструменты (средства визуализации, построения отчетов, статистической обработки и т.д.) и содержимое репозитория, анализирует данные в хранилище. Результатом является информация в виде готовых отчетов, найденных скрытых закономерностей, каких-либо прогнозов.

Так как средства работы конечного пользователя с хранилищем данных могут быть самыми разнообразными, то теоретически их выбор не должен влиять на структуру хранилища и функции его поддержания в актуальном состоянии. Физическая реализация данной концептуальной схемы может быть самой разнообразной.

Изучив данную тему, студент должен:

- знать: основные направления стандартизации технологий БД
- уметь: использовать рекомендации стандартов при исследовании и разработке БД
- иметь представление о: возможностях применения основных стандартов на практике.

При изучении темы необходимо:

- изучить учебный материал: [2, С. 14-23], [3]
- самостоятельно более подробно рассмотреть работы: [4,5].

Для самоконтроля по теме необходимо ответить на следующие вопросы.

1. Предпосылки разработки стандартов технологий баз данных и отдельных их элементов.
2. Дайте характеристику периода становления новых технологий, связанных с созданием, поддержкой и использованием баз данных.
3. Как проходило формирование основ методологии построения систем баз данных.
4. Дайте характеристику формирования теоретических основ современных технологий баз данных.
5. Стандарты баз данных, их назначение и виды. Стандарты открытых систем. Организации по стандартизации технологий баз данных.
6. Стандарт языка SQL
7. ODMG и стандарты объектных баз данных
8. Стандарт ODMG-93
9. Стандарт ODMG 2.0
10. Стандарт ODMG 3.0
11. Стандарты API для систем баз данных
12. Состав информации с точки зрения решения конкретной задачи?
13. Может ли полезная информация быть логически не полной, противоречивой, недостоверной?
14. Как изменяется количество информации, содержащейся в данных, при обработке?
15. Что лежит в основе концепции ХД?
16. Дайте определение ХД (по Инмону).

Вопросы к семинару 1

1. Предмет и назначение системы стандартов по базам данных ГОСТ 34.321-96 «Эталонная модель управления данными».
2. Предмет и назначение системы стандартов ГОСТ 34.320-96 «Концепции и терминология для концептуальной схемы и информационной базы».
3. Предпосылки разработки стандартов технологий баз данных и отдельных их элементов.
4. Период 60-х гг. - становление новых технологий, связанных с созданием, поддержкой и использованием баз данных. Формирование основ методологии построения систем баз данных. Исследование методов организации среды хранения базы данных.
5. Период 70-х гг. - реляционные системы, формирование теоретических основ современных технологий баз данных. Машины баз данных. Развитие методологии моделирования данных.
6. Стандарты баз данных, их назначение и виды. Стандарты открытых систем. Организации по стандартизации технологий баз данных.
7. Стандарт языка SQL
8. ODMG и стандарты объектных баз данных
9. Стандарт ODMG-93
10. Стандарт ODMG 2.0
11. Стандарт ODMG 3.0
12. Стандарты API для систем баз данных.

3.2. Архитектура системы баз данных

3.2.1. Независимость данных

Рассматриваются следующие вопросы: трехуровневая архитектура баз организации базы данных, средства систем управления базами данных для реализации трехуровневой архитектуры.

Архитектура баз данных ANSI/X3/SPARC. Структуры данных, типы данных, модели данных.

Цель изучения: сформировать общее представление о технологии баз данных.

Теоретическая часть

Только к концу 80-х годов была в полной мере осознана необходимость интеграции корпоративной информации и надлежащего управления ею, а также появились технические возможности для создания соответствующих систем, первоначально названных "хранилищами информации" (information warehouse) (Devlin, B.A. and Murphy, P.T. An Architecture for a Business and Information System. IBM Systems Journal. Volume 27, No. 1, 1988), а затем, с выходом книги Инмона, получивших свое нынешнее наименование Хранилищ данных.

Существует два основных подхода к архитектуре Хранилищ данных:

1) корпоративная информационная фабрика (Corporate Information Factory, сокр. CIF) Инмона: скоординированное извлечение данных из источников; загрузка реляционной базы данных с третьей нормальной формой, содержащей атомарные данные; получившееся нормализованное Хранилище используется для того, чтобы наполнить информацией дополнительные репозитории презентационных данных, т.е. данных, подготовленных для анализа: эти репозитории, в частности, включают специализированные Хранилища для изучения и "добычи" данных (Data Mining), а также витрины данных; конечные витрины данных создаются для обслуживания бизнес-отделов или для реализации бизнес-функций и используют пространственную модель для структурирования суммарных данных. Атомарные данные остаются доступными через нормализованное Хранилище данных. Очевидно, что структура атомарных и суммарных данных при таком подходе существенно различается;

2) хранилище данных с архитектурой шины (Data Warehouse Bus, сокр. BUS) Ральфа Кимболла (Ralph Kimball): первичные данные преобразуются в информацию, пригодную для использования, на этапе подготовки данных, при этом обязательно принимаются во внимание требования к скорости обработки информации и качеству данных; подготовка данных начинается со скоординированного извлечения данных из источников (как и в модели Инмона). Ряд операций совершается централизованно, например, поддержание и хранение общих справочных данных, другие действия могут быть распределенными. Область представления пространственно структурирована, при этом она может быть централизованной или распределенной. Пространственная модель Хранилища данных содержит ту же атомарную информацию, что и нормализованная модель (см. подход Инмона), но информация структурирована по-другому, чтобы облегчить ее использование и выполнение запросов. Эта модель включает как атомарные данные, так и обобщающую информацию (агрегаты в связанных таблицах или многомерных кубах) в соответствии с требованиями производительности или пространственного распределения данных. Запросы в процессе выполнения обращаются к все более низкому уровню детализации без дополнительного перепрограммирования со стороны пользователей или разработчиков приложения.

Оба подхода направлены на создание одного объекта - корпоративного Хранилища данных. Соответственно, единство конечного объекта означает общность требований, которым должен удовлетворять любой подход для достижения искомого конечного результата, а это, в свою очередь, указывает на то, что и в самой архитектуре должны быть общие черты.

Первое основное требование связано с тем, что для принятия и осуществления важных решений, от проблем выживания бизнеса до увеличения прибыли, все корпорации нуждаются в программном средстве для хранения, анализа и интерпретации данных, которые они накапливают. Для достижения своих целей корпорации должны в полной мере использовать возможности первичных данных, что, в свою очередь, требует

соответствующих операционных систем и систем обратной связи. Именно для этого создаются Хранилища данных, и оба архитектурных подхода в полной мере отвечают названным требованиям.

Второе требование - это требование точности и своевременности данных. Каждый пользователь должен иметь возможность доступа к любым данным в соответствии со своими конкретными требованиями, и этот доступ должен осуществляться с помощью легко понятных и простых способов построения запросов. Как Хранилище данных с архитектурой шины (BUS), так и корпоративная информационная фабрика (CIF) вполне удовлетворяют и второй группе требований.

В техническом отношении оба подхода признают необходимость наличия в Хранилище как атомарных, так и суммарных данных, а также то, что данные обеих категорий должны быть доступны пользователям для анализа.

Рассмотрим различия в предлагаемых подходах.

Первое - различные подходы к построению баз данных, составляющих основу Хранилища. Архитектуры отличаются только способами обращения с атомарными данными: пространственная организация (Кимболл), нормализованная (Инмон). Ральф Кимболл (Ralph Kimball) использует пространственную организацию баз данных (dimensional data bases) с так называемой архитектурой «звезда» как на стадии подготовки, так и презентации данных. Инмон (W. Inmon) комбинирует два подхода. В его модели атомарные данные организованы в реляционные базы и находятся в нормализованном Хранилище данных, причем суммарные данные доступны для использования через специализированные Хранилища, средства data mining и OLAP; что же касается зависимых витрин данных, то только они организованы с помощью пространственных моделей, как и у Ральфа Кимболла.

Изучив данную тему, студент должен:

- *знать:* основные понятия теории баз данных;
- *уметь:* обосновать выбор структуры данных, модели данных
- *иметь представление об:* архитектуре системы базы данных (на основе трехсхемной архитектуры ANSI/X3/SPARC).

При изучении темы необходимо:

- изучить учебный материал: [2, С. 23-37, 42-50]
- самостоятельно более подробно рассмотреть работы: [4,5]

Особое внимание обратить на: архитектуру системы базы данных на основе трехсхемной архитектуры ANSI/X3/SPARC [2, С.23-29]

Для самоконтроля по теме необходимо ответить на следующие вопросы.

1. Охарактеризуйте понятия: база данных, схема базы данных, метаданные, словарь данных.
2. Что такое жизненный цикл баз данных? Этапы ЖЦ.
3. Дайте характеристику архитектуры ANSI/SPARC.
4. Опишите три уровня абстракции данных.
5. Какие требования предъявляются к организации баз данных?
6. Что такое структурирование?

7. Дайте характеристику архитектуры Хранилищ данных. Приведите примеры основных подходов в построении Хранилищ данных.

8. В каком подходе будет верным утверждение, что структуры атомарных и суммарных данных существенно различаются?

9. В чем сходство известных подходов к архитектуре Хранилищ данных?

10. В чем различия известных подходов к архитектуре Хранилищ данных?

Содержание лабораторных занятий.

ЛР 3. На основе результатов анализа предметной области (лабораторные работы 1 и 2) построить функциональную модель бизнес процесса обработки информации в нотации DFD. Модель должна быть представлена, как минимум, двумя диаграммами: контекстной и главной (А-0 и А0). Выполнить документирование модели (название, цель, точка зрения, функции обработки информации, информационные потоки (входные, выходные), хранилища данных, внешние сущности).

Пример выполнения заданий лабораторных работ представлен в приложении 2.

3.2.2. Модели данных

Рассматриваются следующие вопросы: преимущества централизованного управления данными, современные тенденции построения файловых систем, выбор модели данных, иерархическая, сетевая и реляционная модели данных, их типы структур, основные операции и ограничения

Цель изучения: сформировать представление о моделях данных.

Методология создания систем баз данных, модели данных, среда хранения данных. Сетевой, иерархический и реляционные подходы к построению баз данных.

Постановка задачи. В гипотетическом пункте обмена валют создается локальная информационная система (ИС), призванная автоматизировать процесс учета сделок купли-продажи валюты. Создаваемая система должна обеспечить ввод, хранение и поиск информации о сделках, совершенных в данном пункте обмена валют. Каждой сделке присваивается уникальный цифровой код. Информация о сделке должна содержать сведения о дате, времени сделки, суммах покупаемой и продаваемой валют, фамилии, имени, отчестве и номере паспорта клиента, а также о фамилии, инициалах и учетном номере личного дела кассира в отделе кадров. Система должна позволять вычислить денежный оборот за один или несколько дней, а также осуществлять поиск информации о сделках по номеру паспорта клиента. Задача состоит в проектировании структуры базы данных разрабатываемой автоматизированной ИС.

Создание логической модели БД

Проведем анализ предметной области с целью выделить основные сущности. Поскольку речь идет об учете сделок купли-продажи валюты, ясно,

что в модели должна присутствовать сущность СДЕЛКА. Как вы помните, согласно правилам, на ER-диаграмме названия сущностей записываются большими буквами. Понятие сделки подразумевает участие по крайней мере двух совершающих ее сторон, а также наличие предмета сделки, поэтому участниками сделки являются клиент и кассир, а предметом сделки является валюта, необходимо ввести еще три сущности: ВАЛЮТА, КЛИЕНТ, и КАССИР.

Определение связей между сущностями. Для задания связей между сущностями сначала составим описание данной предметной области при помощи ряда истинных высказываний на естественном языке.

Любой КЛИЕНТ должен совершать

Каждую СДЕЛКУ должен совершать

Любой КАССИР может обслуживать

Каждую СДЕЛКУ должен обслуживать

Любая ВАЛЮТА может покупаться и продаваться

При совершении СДЕЛКИ должна покупаться одна ВАЛЮТА и продаваться другая ВАЛЮТА.

Анализ приведенных высказываний позволяет выделить четыре связи (названия связей - глаголы):

КЛИЕНТ совершает СДЕЛКУ.

КАССИР обслуживает СДЕЛКУ.

ВАЛЮТА покупается при СДЕЛКЕ.

ВАЛЮТА продается при СДЕЛКЕ.

Во всех четырех случаях связи являются связями "один - ко - многим", связи неидентифицирующие. Так как любой экземпляр сущности СДЕЛКА может быть однозначно идентифицирован по коду сделки, то есть вне зависимости от экземпляров других сущностей. Все связи, все кроме первой могут иметь мощность 0,1 или более. Первая связь не может иметь мощность 0, так как в данном случае любой человек становится КЛИЕНТОМ только тогда, когда он совершает хотя бы одну сделку.

Указание первичных ключей и неключевых атрибутов. Сведения о клиенте должны состоять из его фамилии, имени, отчества и номера его паспорта. Введем для каждого КЛИЕНТА уникальный номер, который и будет первичным ключом. А атрибут "номер паспорта клиента" сделаем альтернативным ключом, чтобы обеспечить возможность быстрого поиска информации о сделках по его значениям. Атрибутами сущности КАССИР будут фамилия, инициалы и учетный номер кассира. Поскольку учетный номер личного дела кассира может содержать не только цифры, но и буквы, и, следовательно, для его хранения будет использоваться строка минимум из 13 символов, что не очень удобно, введем для каждого экземпляра уникальный номер, который и будет первичным ключом. Сущность ВАЛЮТА будет содержать два атрибута: код валюты и название валюты, первый из которых будет первичным ключом. Что касается сущности СДЕЛКА, то часть атрибутов она унаследует от родительских сущностей, и остается лишь добавить следующие: «дата сделки», «время сделки», «сумма покупаемой валюты» и

«сумма продаваемой валюты». Очевидно, что первичным ключом, следует выбрать уникальный цифровой код сделки.

Создаваемая система должна позволять вычислить денежный оборот за один или несколько дней, полезно было бы сделать атрибут «дата сделки» инверсным входом, так как он будет часто использоваться для доступа к данным.

Инверсный вход – атрибут или группа атрибутов, которые не определяют экземпляр сущности уникальным образом, но часто используются для обращения к экземплярам сущности.

Изучив данную тему, студент должен:

- *знать:* основные логические модели данных
- *уметь:* графически отображать семантику предметной области средствами моделирования
- *иметь представление о:* семантике предметной области

При изучении темы необходимо:

- изучить учебный материал: [2, С. 42-50, 190-197, 200-214]
- самостоятельно более подробно рассмотреть работы: [5, <http://www.citforum.ru/index.html>]

Особое внимание обратить на: понимание сути семантического моделирования.

Для самоконтроля по теме необходимо ответить на следующие вопросы.

1. Характеристика модели «сущность-связь».
2. Как осуществляется поддержание целостности базы данных?
3. Что из себя представляют концепция и семантика методики IDEF1X?
4. Каким образом производится идентификация сущностей в IDEF1X?
5. Какие вы знаете инструменты поддержки стандартов моделирования?
6. Каким требованиям должен удовлетворять современный инструмент моделирования баз данных?
7. Дайте характеристику AllFusion ERwin Data Modeler.

Вопросы к семинару 2

1. Основные функции наиболее известного CASE-средства функционального моделирования? Назначение методологии диаграмм потоков данных.
2. Какими правилами регламентируется построение диаграммы DFD при описании документооборота и обработки информации? В чем особенности DFD-диаграмм? Основные компоненты диаграммы DFD. В чем особенности объектов DFD-диаграмм?
3. Какие элементы процессов документооборота и обработки информации могут быть отображены на диаграммах DFD? В чем состоит назначение процесса? Что называется внешней сущностью? Что такое поток данных? Какова функция хранилища данных в DFD?
4. Характеристика ERwin Data Modeler.

5. Создание логической модели данных.
6. Основные компоненты логической модели. Характеристика сущности: типы, наследование (иерархия категорий) и т.д..
7. Основные компоненты логической модели. Характеристика атрибута.
8. Основные компоненты логической модели. Характеристика связи: мощность, именование, тип и т.д.
9. Определение ключевых атрибутов. Характеристика РК, АК, IE, FK.
10. Определение правил ссылочной целостности.
11. Нормализация данных: определение, типы зависимости, 1NF, 2NF, 3NF.
12. Определение домена.

Содержание лабораторных занятий.

ЛР 4. Для отображения внешней модели (функциональная модель обработки информации - DFD модель) на концептуальную модель необходимо привести в соответствие информационные объекты DFD модели и объекты информационной модели.

Задание. Определение объектов предметной области и их характеристик:

1. Определить основные ограничения и допущения относительно будущей системы;
2. Выделить информационные объекты функциональной модели обработки информации (DFD модель, см. ЛР 3);
3. Выполнить анализ структуры информационных объектов функциональной модели обработки информации (DFD модель, см. ЛР 3) (информационные потоки, хранилища данных);
4. Построить таблицу соответствия информационных объектов DFD модели и их характеристик: поля-названия информационных объектов, строки – элементы структуры (характеристики) информационного объекта, если элемент имеется в нескольких информационных объектах, ставим знак «+» в ячейку пересечения;
5. Эмпирическим путем (по смыслу) определить взаимосвязи между элементами информационного объекта-сущности;
6. Определить информационные объекты предметной области;
7. Выполнить нормализацию.

Пример выполнения заданий лабораторных работ представлен в приложении 3.

ЛР 5. Построение концептуальной схемы базы данных. На основании выявленных и определенных информационных объектов, их характеристик и взаимосвязей (см. ЛР 4), построить информационную модель (ERD-модель) в соответствии с методологией IDEF1X с использованием любого инструментального средства.

Вспомнить методологию семантического моделирования данных IDEF1X

Пример выполнения заданий лабораторных работ представлен в приложении 4.

ЛР 6. Обоснование выбора СУБД для реализации реляционной базы данных.

Задание

Для дальнейшего проектирования и реализации базы данных необходимо выбрать СУБД, в среде которой и будет создана база данных, и выполнить обоснование выбора по предлагаемой ниже методике. Обязательно определить параметры для 50% критериев из группы, определяющих выбор СУБД.

Выполнить заполнение таблицы значениями параметров группы критериев, определяющих выбор СУБД.

СУБД для выбора: MS Access, Oracle Database, Firebird, Interbase, IBM DB2, Informix, MS SQL Server, Sybase Adaptive Server Enterprise, PostgreSQL, MySQL и другие (на ваш выбор).

Теоретический материал

Следует отметить, что по существующей практике решение об использовании той или иной СУБД принимает один человек - обычно, руководитель предприятия, а он может опираться отнюдь не на технические критерии. Здесь свою роль могут сыграть такие, с технической точки зрения, незначительные факторы как рекламная раскрутка компании-производителя СУБД, использование конкретных систем на других предприятиях, стоимость. При этом последний фактор может трактоваться в двух противоположных смыслах в зависимости от финансового состояния и политики предприятия. С одной стороны, это может быть принцип, - чем дороже, тем лучше. С другой стороны - культивирование почти бесплатного использования продукта, вплоть до взлома его лицензионной защиты. Очевидно, последний подход чреват коллизиями и не может привести к успеху в долгосрочной работе.

Группы критериев, определяющих выбор СУБД

1. Моделирование данных.
2. Особенности архитектуры и функциональные возможности.
3. Контроль работы системы.
4. Особенности разработки приложений.
5. Производительность.
6. Требования к рабочей среде
7. Надежность.
8. Смешанные критерии.

Каждый из этих критериев может быть охарактеризован через набор определенный параметров. Четкий и глубокий сравнительный анализ на основании перечисленных критериев в любом случае поможет рационально выбрать подходящую систему для конкретного проекта, и затраченные усилия не будут напрасными. Перечень критериев поможет осознать масштабность задачи и выполнить ее адекватную постановку.

Пример выполнения заданий лабораторных работ представлен в приложении 5.

3.3. Защита данных в системах с базами данных

3.3.1. Реализация защиты данных

Рассматриваются следующие положения: программное обеспечение RAID, аппаратное устройство RAID, компоненты RAID-массива, RAID-уровни, сравнение RAID-конфигураций.

Цель изучения: сформировать представление об организации процессов обработки данных.

Теоретическая часть

Администрирование баз данных; управление параллельной обработкой; безопасность базы данных; восстановление базы данных, ограничения целостности, физическая и логическая целостность баз данных. Способы обработки данных: централизованная, децентрализованная, распределенная, интегрированная.

Централизованная обработка данных предполагает наличие вычислительного центра (ВЦ) При этом способе пользователь доставляет на ВЦ исходную информацию и получают результаты обработки в виде результативных документов. Особенностью способа обработки являются сложность и трудоемкость налаживания быстрой, бесперебойной связи, большая загруженность ВЦ информацией, регламентацией сроков выполнения операций, организация безопасности системы от возможного несанкционированного доступа.

Децентрализованная обработка данных. В настоящее время существуют три вида технологий децентрализованной обработки данных: Первая: основывается на ПК, не объединенных в локальную сеть (данные хранятся в отдельных файлах и на отдельных дисках). Для получения показателей производится перезапись информации на компьютер. Вторая: ПК объединенные в локальную сеть, что ведет к созданию единых файлов данных (но он не рассчитан на большие объемы информации). Третья: ПК объединенные в локальную сеть, в которую включаются специальные серверы (с режимом «клиент- банк»).

Распределенный способ обработки данных основан на распределении функций обработки между различными ЭВМ, включенными в сеть. Этот способ может быть реализован двумя путями: установка ЭВМ в каждом узле сети, при этом обработка данных осуществляется одной или несколькими ЭВМ в зависимости от реальных возможностей системы и ее потребностей на текущий момент времени, размещение большого числа различных процессоров внутри одной системы. Такой путь применяется в системах обработки банковской и финансовой информации, там, где необходима сеть обработки данных (филиалы, отделения и т.д.).

Интегрированный способ обработки данных предусматривает создание информационной модели управляемого объекта, то есть создание распределенной базы данных. Такой способ обеспечивает максимальное удобство для пользователя. С одной стороны, базы данных предусматривают

коллективное пользование и централизованное управление. С другой стороны, объем информации, разнообразие решаемых задач требуют распределения базы данных. Технология интегрированной обработки информации позволяет улучшить качество, достоверность и скорость обработки, т.к. обработка производится на основе единого информационного массива, однократно введенного в ЭВМ. Особенностью этого способа является отделение технологически и по времени процедуры обработки от процедур сбора, подготовки и ввода данных.

Преимущества технологии распределенной обработки данных большое число взаимодействующих пользователей, выполняющих функции сбора, регистрации, хранения, передачи и выдачи информации; снятие пиковых нагрузок с централизованной базы путем распределения обработки и хранения локальных баз данных на разных ЭВМ; обеспечение доступа информационному работнику к вычислительным ресурсам сети ЭВМ; обеспечение симметричного обмена данными между удаленными пользователями.

Технология распределенной обработки данных. В системе распределенной обработки клиент может послать запрос к собственной локальной или удаленной базе данных. Удаленный запрос -это единичный запрос к одному серверу. Несколько удаленных запросов к одному серверу объединяются в удаленную транзакцию. Если отдельные запросы транзакции обрабатываются различными серверами, то транзакция называется распределенной. При этом один запрос транзакции обрабатывается одним сервером. Распределенная СУБД позволяет обрабатывать один запрос несколькими серверами. Такой запрос называется распределенным. Только обработка распределенного запроса поддерживает концепцию распределенной базы данных. Организация обработки данных зависит от способа распределения.

Технология «файл – сервер». Сетевые СУБД, основанные на файл-сервере, в настоящее время недостаточно мощны. В нагруженной сети неизбежно падает производительность, нарушается безопасность и целостность данных. Проблема производительности возникла потому, что файл-серверы реализуют принцип «все или ничего». Полные копии файлов базы перемещаются взад-вперед по сети. Проблемы с безопасностью, целостностью возникли из-за того, что с самого начала файл-серверы не были сконструированы с учетом целостности данных и их восстановления в случае аварии. В базах данных коллективного пользования центральным технологическим звеном становятся серверы баз данных. Программные средства серверов баз данных обеспечивают реализацию многопользовательских систем, централизованное хранение, поиск и обработку, целостность и безопасность данных. Производительность серверов баз данных на порядок выше производительности файл-серверов. В отличие от файл-сервера сервер базы данных содержит и базу, и систему управления данными.

Технология «клиент – сервер». Основная идея технологии "клиент-сервер" заключается в том, чтобы расположить серверы на мощных машинах, а приложения клиентов, использующих язык инструментальных средств, - на

менее мощных машинах. Тем самым будут задействованы ресурсы более мощного сервера и менее мощных машин клиентов. Ввод-вывод к базе основан не на физическом дроблении данных, а на логическом, т. е. клиентам отправляется не полная копия базы, а сервер посылает только логически необходимые порции, тем самым сокращая трафик сети. Трафик сети - это поток сообщений сети. В технологии "клиент-сервер" программы клиента и его запросы хранятся отдельно от СУБД. Сервер обрабатывает запросы клиентов, выбирает необходимые данные из базы данных, посылает их клиентам по сети, производит обновление информации, обеспечивает целостность и сохранность данных.

Способы соединения программ клиента и сервера баз данных Во всех технологиях существуют два способа связи прикладных программ клиента и сервера баз данных: Прямой - прикладная программа клиента связывается непосредственно с сервером базы данных Непрямой - доступ к удаленному серверу обеспечивается средствами локальной базы Возможно объединение обоих способов.

Изучив данную тему, студент должен:

- *знать:* основные принципы организации процессов обработки данных.
- *уметь:* описать основные этапы технологии оперативной обработки транзакций – OLTP-технологии.

- *иметь представление о:* процессах администрирования баз данных

При изучении темы необходимо:

- изучить учебный материал: [2, С. 54-61, 89-103]

- самостоятельно более подробно рассмотреть работы: 5, 6, 7.

Особое внимание обратить на: вопросы процесса администрирования БД, понимание сути транзакции.

Для самоконтроля по теме необходимо ответить на следующие вопросы.

1. Дайте характеристику этапам развития средств хранения и доступа к информации.
2. Дайте обоснование многоуровневости архитектуры СУБД.
3. Приведите в соответствие уровни описания данных и уровни архитектуры СУБД.
4. Какие функциональные возможности должна иметь современная СУБД?
5. Каким образом реализуются основные функции СУБД?

Содержание лабораторной работы.

ЛР 7. Данная лабораторная работа посвящена созданию запросов. Рассмотрим еще один режим работы с запросами – режим SQL. Нужно понимать, что после создания даже с помощью мастера или в режиме конструктора, запрос в базе данных хранится как выражение на языке SQL. Знание SQL нередко позволяет быстрее написать сложные запросы или разобраться при возникновении каких-то проблем.

Задание 1.

Используя базу lib.accdb в конструкторе создайте запрос, выводящий список авторов книг из таблицы Book (нам нужны только авторы, без названий

книг и т.д.). Добавьте в таблицу данные таким образом, чтобы в ней было несколько книг одного автора. Выполните запрос и обратите внимание на то, что если автор написал несколько книжек, то в результатах запроса он будет фигурировать тоже несколько раз. Перейдите в режим SQL, в котором Вы увидите то предложение SQL, которое и является запросом:

SELECT Book.Author FROM Book

В SQL для указания на то, что нужно отбрасывать повторения, служит ключевое слово DISTINCT. Измените запрос так, как показано ниже и проверьте его работу.

SELECT DISTINCT Book.Author FROM Book

Нужно отметить, что в выражении SELECT можно явно указать, что надо возвращать все записи, включая дубликаты. Для этого используется ключевое слово ALL, но обычно его не ставят, т.к. оно подразумевается, если ничего другого явно не указано. Обратите внимание, что при выполнении второго запроса, кроме отбрасывания повторяющихся записей, может поменяться порядок их вывода.

Если требуется задать определенный порядок сортировки выводимых записей, в режиме SQL это можно сделать, добавив инструкцию ORDER BY, после указывается столбец или столбцы, по 2 которым проводится сортировка. По умолчанию или при явном добавлении ключевого слова ASC сортировка проводится по возрастанию, если стоит DESC – по убыванию.

В приведенных ниже примерах первый запрос выводит список авторов без повторений в алфавитном порядке. Второй запрос выводит без повторения перечень авторов и годов издания книг, причем сортировка по первому столбцу производится по возрастанию, а среди записей с одинаковым значением первого поля, сортировка производится по году издания книг по убыванию.

SELECT DISTINCT Book.Author FROM Book ORDER BY Book.Author;
SELECT DISTINCT Book.Author, Book.BookYear FROM Book ORDER BY Book.Author ASC, Book.BookYear DESC

Задание 2.

Напишите запрос, выводящий информацию о книгах из таблицы Book в порядке от самых старых к самым новым. Книги с одинаковым годом издания упорядочивайте в списке по названию в алфавитном порядке. В некоторых случаях нужно вывести не весь сортированный список, а какую-то его часть. Например, нас могут интересовать первые 5 записей из списка или первые 10 процентов списка. В подобных случаях, в запрос SELECT добавляют соответственно «TOP n» или «TOP n PERCENT», где n – количество записей или процент.

Например, следующий запрос выведет список из двух самых новых изданий из Book (символ * указывает на то, что нам нужны все столбцы).

*SELECT TOP 2 * FROM Book ORDER BY Book.BookYear DESC*

Задание 3.

Напишите запрос, выводящий информацию книгах, составляющих первую половину общего списка из таблицы Book, упорядоченного по возрастанию года (т.е. самые старые книги). В конце предыдущей лабораторной мы

познакомились с запросами, использующими данные из нескольких связанных таблиц.

Например, построенный в конструкторе запрос на выборку 3 библиотечных номеров книг, их авторов и названий, на SQL будет выглядеть следующим образом:

```
SELECT BookInLib.LibID, Book.Author, Book.Title FROM Book INNER JOIN BookInLib ON Book.BookID = BookInLib.BookID
```

Здесь инструкция INNER JOIN указывает на то, что надо соединить таблицы по условию совпадения значений в столбцах Book.BookID и BookInLib.BookID. Причем соединение должно быть «внутренним» – соответствующие значения должны встречаться как в одной, так и в другой таблице. Кроме внутреннего, Access также поддерживает левое и правое внешнее соединение – LEFT JOIN и RIGHT JOIN. В первом случае в выборку берутся все записи из левой таблицы и соответствующие им записи из правой, если они есть. Во втором случае – наоборот.

Задание 4.

Скопируйте приведенный выше запрос в свою базу. Выполните его. При необходимости измените набор данных в таблицах так, чтобы в таблице Book были записи не связанные с записями из таблицы BookInLib. В запросе замените INNER JOIN на LEFT JOIN, проанализируйте результат. Объясните, почему для нашей структуры базы данных замена в этом запросе INNER JOIN на RIGHT JOIN не приведет к изменениям в выводимом наборе данных (тут надо вспомнить о внешних ключах, которые создавались в ходе выполнения первой лабораторной).

В предложениях SELECT, условия, накладываемые на отбираемые записи, указываются после ключевого слова WHERE.

Например, запрос на выборку книг, выпущенных с 1997 по 2001 год: *SELECT Book.Author, Book.Title, Book.Publisher, Book.BookYear FROM Book WHERE Book.BookYear BETWEEN 1997 and 2001*

Условия могут включать проверку на равенство, неравенство, принадлежность множеству (IN), соответствие текстовой маске (LIKE) и т.д. Условие также может быть составным и включать логические операторы NOT (отрицание), AND (и), OR (или); их приоритет убывает именно в такой 4 последовательности. Например, пусть кроме условия на год издания, надо проверить принадлежность издательства заранее определённого списку.

Запрос может выглядеть следующим образом.

```
SELECT Book.Author, Book.Title, Book.Publisher, Book.BookYear FROM Book WHERE Book.BookYear BETWEEN 1997 AND 2001 AND Book.Publisher IN ('Азбука', 'Политехника')
```

Эквивалентная запись этого запроса может быть такой:

```
SELECT Book.Author, Book.Title, Book.Publisher, Book.BookYear FROM Book WHERE Book.BookYear BETWEEN 1997 AND 2001 AND (Book.Publisher = 'Азбука' OR Book.Publisher = 'Политехника')
```

Обратите внимание, что если скобки в условии WHERE не поставить, то логика работы запроса изменится: он будет выводить книги, изданные

издательством «Азбука» в указанном диапазоне лет, и любые книги издательства «Политехника». Поэтому очень важно помнить про порядок выполнения операций NOT, AND, OR и правильно использовать скобки в случаях, подобных рассмотренному выше.

Задание 5.

Напишите на SQL запрос, который выведет в обратном алфавитном порядке названия книг, выпущенных после 1999 года, фамилия автора (значение поля Author) у которых начинается с буквы «Г» или название издательства заканчивается буквой «а».

Напишите на SQL запрос, возвращающий список названий неиспользуемых статусов книг, то есть таких, что данный статус не назначен ни одной книге в таблице BookInLib.

Подсказка: нужно использовать внешнее соединение таблиц и условие IS NULL. При необходимости отредактируйте используемый набор данных, чтобы такие статусы были, и проверьте работу запроса.

Задание 6.

Опишите, что на ваш взгляд делает приведённый ниже запрос. Скопируйте его код в Access и выполните. Правильным ли оказалось ваше предположение?

`SELECT * FROM Book WHERE Author LIKE 'A*' OR BookYear > 2000 AND NOT Publisher LIKE '[И,П]*'`

Пример выполнения заданий лабораторных работ представлен в приложении 6.

3.4. Управление системами хранения данных

3.4.1. Интеллектуальные системы хранения данных

Рассматриваются следующие положения: компоненты интеллектуальной системы хранения данных, интеллектуальный массив хранения данных, концепции на практике.

Цель изучения: знакомство с методологией многомерного моделирования данных, изучить основные подходы к моделированию с использованием различного инструментария, познакомить с этапами проектирования хранилищ данных.

Теоретическая часть

В течение последнего десятилетия термин DataWarehouse или Хранилище Данных стал одной из самых популярных тем для обсуждения. Любая система, хранящая собираемые данные, претендует на название Хранилища Данных. Ведутся ожесточенные теоретические споры по поводу, что называть Хранилищем Данных, чем оно отличается от OLTP систем и какое определение более точно описывает его сущность. Мы попробуем подойти с другой стороны – практической.

Одним из самых важных вопросов при построении любой информационной системы является четкое представление о том, что и как должно строиться, а говоря более строгим языком, – архитектура системы и

методология ее построения. Для информационных систем в технологии Хранилищ Данных, главными особенностями которых являются большой масштаб проекта и постоянное развитие, наличие этих составляющих является обязательным.

Методология. Любой проект, в конечном счете, должен принести прибыль. По общему мнению, Хранилище Данных создается для информационной поддержки процесса принятия решений, осуществляет процесс доставки необходимой, актуальной и верной информации нужным людям в нужное время, с тем, чтобы они могли принимать обоснованные и своевременные решения. Для большой организации велико число лиц, принимающих решения и нуждающихся в такой информационной поддержке. Реализация проекта, охватывающего деятельность всей организации, потребует длительного времени и больших затрат. К тому же, конкретные требования к доставляемой информации меняются достаточно быстро, и возможно, что по окончании проекта его актуальность будет далека от ожидаемой. Осознание этого факта приводит нас к двум выводам. Во-первых, создание ХД должно осуществляться короткими и быстро дающими ощутимые результаты этапами. Во-вторых, созданное Хранилище Данных не будет статической, раз и навсегда созданной системой. Требования к доставляемой для принятия решений информации будут меняться в процессе его эксплуатации и развития, поэтому логическая и физическая структура ХД, должна позволять достаточно быстро и безболезненно осуществлять требуемые изменения.

Думать стратегически, начинать с решения самых насущных задач – вот лозунг, под которым идет создание и развитие Хранилища Данных. Практически это означает, что на первой стадии важно создать ясную картину будущего содержания Хранилища Данных в самых грубых чертах, затем выбрать наиболее актуальную задачу, и руководствуясь концепцией архитектуры Хранилища Данных работать над ней, не забывая об общей картине.

Этот подход включает в себя определение стратегических путей развития на самых ранних этапах построения Хранилища и поэтапную реализацию этих планов с учетом заданных приоритетов в виде краткосрочных (4-8 месяцев) проектов. Такой подход позволяет получить ощутимые результаты в очень короткие сроки и при этом сохранить уверенность в том, что дальнейшее расширение создаваемого Хранилища не приведет к перестройке всей его структуры.

Оценка проекта. С самого начала необходимо оценить условия для реализации проекта, которые помимо традиционных условий, таких как достаточный бюджет, наличие команды разработчиков и прочие, включают понимание концепции Хранилищ Данных, определенный уровень информационной инфраструктуры и другие. Для эффективного контроля над ходом выполнения проекта важно также с самого начала выбрать четкий и ясный критерий оценки проекта.

Изучив данную тему, студент должен:

- *знать*: основные принципы организации процессов обработки данных.

- *уметь*: описать основные этапы технологии оперативной обработки транзакций – OLTP-технологии.

- *иметь представление о*: процессах администрирования баз данных

При изучении темы необходимо:

- изучить учебный материал: [2, С. 54-61, 89-103]

- самостоятельно более подробно рассмотреть работы: 5, 6, 7.

Особое внимание обратить на: вопросы процесса проектирования хранилища данных.

Для самоконтроля по теме необходимо ответить на следующие вопросы.

1. Каким образом организуются данные при использовании РСУБД для организации ХД?

2. Дайте характеристику размерной модели типа «звезда».

3. Дайте характеристику компонентам модели «звезда».

4. Что можно предпринять для увеличения производительности анализа в модели «звезда»?

5. Дайте характеристику компонентам схемы «снежинка».

6. С какой целью в состав реляционных СУБД включают оптимизаторы запросов?

7. Преимущества и недостатки использования многомерной модели при построении ХД.

8. Преимущества и недостатки использования реляционной модели при построении ХД.

9. Как называется специализированное тематическое хранилище, обслуживающее одно из направлений деятельности организации?

10. На основе какой архитектуры обычно строятся системы, использующие хранилище данных?

Содержание лабораторной работы

ЛР 8. Построение многомерной модели хранилища данных для киоска данных.

Описание предметной области. Компания «Mebele_Life» занимается производством и реализацией мебели (мебельных комплектующих). Киоск данных используется аналитиками для детального изучения взаимосвязи расходов и доходов компании от реализации продукции и подготовки отчетности о продажах для руководства.

Компания открыла сеть точек продаж, которые находятся в городах Екатеринбург и Магнитогорск, и так же осуществляют производственную деятельность, т. е. сделала расширение своей деятельности. Руководство компании хочет оценить эффективность сделанного расширения и иметь более подробную информацию о зависимости между продажами и производством по затратам и доходам.

Компания выпускает около 200 видов (моделей) мебельной продукции. Каждый продукт имеет базовый набор комплектующих компонентов. Дополнительные комплектующие компоненты используются для создания специфической (уникальной) модели продукта. Политика компании строится

таким образом, что число выпускаемых моделей остается постоянным. Это означает, что количество новых моделей приблизительно равно количеству моделей, снятых с производства.

Для каждой модели каждого продукта в зависимости от спроса применяется гибкая система скидок. Как правило, размер скидки для покупателей больших партий продукции определяет заведующий складом розничной продажи. Когда принято решение приостановить производство продукции данной модели, информация о ней сохраняется в базе данных компании в течение 12 мес. после того, когда вся оставшаяся продукция будет реализована или списана. Данные о продукции удаляются в тот момент, когда удаляются данные о последней модели этой продукции.

Склад продает за наличный расчет. Независимо от предоставления скидок цена товара меняется. Хотя на каждую продажу продукции оформляется счет, компания не ведет учет покупателей для продаж.

Киоск данных компании предназначен для решения задач анализа показателей расхода и дохода. Типовые запросы, на которые система должна давать ответы, следующие:

1. Какова величина общих издержек и общей прибыли по каждой модели товара, проданной сегодня и просуммированной по точкам продажи, типу точки продажи, по региону и по складам торговли?

2. Какова величина общих издержек и общей прибыли для каждой модели товара, проданной сегодня и просуммированной по заводам и по регионам?

3. Какой процент моделей получили скидки, и какие из них были проданы по факту со скидкой (в процентах) в складах для всех продаж на этой неделе? В этом месяце?

4. Для каждой модели товара, проданной в текущем месяце, определить, какой был процент продаж с торговли по безналичному расчету?

5. Какие модели и какого типа не продавалась в течение последнего месяца? В течение последней недели?

6. Какие 5 моделей, проданных за последний месяц, принесли наибольшую прибыль? По продажам за квартал? По всем продажам?

Источником данных для киоска данных является фрагмент БД системы оперативной обработки данных «Mebele_Life».

Задание.

Построить многомерную модель хранилища данных киоска данных для анализа продаж компании «Mebele_Life» (звезда, снежинка) с использованием любого инструментария.

Пример выполнения заданий лабораторных работ представлен в приложении 7.

3.4.2. Сети хранения данных

Рассматриваются следующие положения: сети хранения данных и их эволюция, компоненты, способы подключения, архитектура, топология, концепции на практике.

Цель изучения: сформировать представление об организации процессов обработки данных.

Теоретическая часть

Метаданные (или данные о данных) являются ключевым элементом в Хранилище Данных. Именно благодаря использованию Метаданных Хранилище становится гибким и удобным средством доставки информации для поддержки принятия решений. Они содержат: полное описание логической и физической структур данных, всех процессов загрузки данных, специализированных приложений для анализа и представления данных в определенных областях, дополнительную информацию обо всех элементах Хранилища, помогающую легко ориентироваться в его сложной структуре.

Метаданными явно или скрыто пользуются все группы пользователей Хранилища, начиная от наименее подготовленных конечных пользователей, приложения для которых управляются Метаданными, до Администратора данных и разработчиков. По функциональным требованиям эти средства можно разделить на две основные группы: средства просмотра и поиска, средства создания и поддержки.

Система SAS предлагает средство MetaSpace Explorer для осуществления просмотра и поиска Метаданных Хранилища для конечных пользователей и имеет следующую функциональность: навигация по объектам Метаданных Хранилища в разрезах: предметная область, тип, владелец и др.; навигация по распределенному Хранилищу в разрезе серверов; поиск объектов Хранилища Данных по заданному критерию; отображение метаданных, связанных с объектами.

В качестве рабочего инструмента Администратора Хранилища Данных и разработчиков создан продукт SAS/Warehouse Administrator, позволяющий осуществлять следующие операции: определять объекты Хранилища, их атрибуты и взаимосвязи; задавать доступ к внешним источникам; описывать процедуры загрузки в Хранилище; задавать физическую модель Хранилища; выполнять процедуры загрузки; регламентировать выполнение процедур загрузки; генерировать метаданные автоматически; представлять структуру Хранилища в удобной графической форме.

Продукт SAS/Warehouse Administrator состоит из следующих основных компонент:

- Warehouse Explorer – проводник по Хранилищу;
- Process Editor – редактор процессов;
- Scheduler – диспетчер процессов.

Warehouse Explorer – проводник по Хранилищу, позволяет: создавать, редактировать, удалять, группировать, просматривать основные объекты Хранилища Данных. В их состав входят: описания внешних источников данных, предметные области и их логические структуры, таблицы детальных данных, таблицы агрегированных данных и MDDB, витрины данных, информационные витрины и приложения, а также вспомогательные объекты. Также это средство позволяет задавать основные атрибуты Хранилища:

рабочие библиотеки, сервера данных и доступ к ним, лиц, ответственных за эксплуатацию и разработку элементов Хранилища.

Process Editor – редактор процессов позволяет: задавать процедуры загрузки для всех элементов Хранилища, от процедур выборки данных из внешних источников до обновления информации в Информационных Витринах и Витринах Данных, атрибуты выполнения этих процедур, задавать связи между объектами Хранилища, графически представлять процесс загрузки любого объекта Хранилища в виде дерева процессов.

Scheduler – диспетчер процессов позволяет регламентировать выполнение процессов загрузки Хранилища Данных. Включает поддержку распределенных Хранилищ Данных.

Продукт SAS/Warehouse Administrator имеет следующие возможности для расширения функциональности и настройки под конкретный проект:

1. MetaData API – программный интерфейс доступа к Метаданным Хранилища;
2. Scheduler API – программный интерфейс управления диспетчером загрузки;
3. Средства расширения функциональности – стандартный интерфейс вызова, дополнительных средств работы с Хранилищем Данных.

Продукт SAS/Warehouse Administrator имеет следующие возможности для расширения функциональности и настройки. Ряд весьма полезных дополнительных средств расширения функциональности имеется в Общем Фонде SAS средств, который пополняется разработками из реальных проектов по всему миру и включает такие полезные средства: средство автоматического создания документации Хранилища Данных, импорт описания структур данных из разнообразных CASE средств и другие.

Загрузка. Для принятия обоснованных решений необходимо, чтобы доставляемая информация была актуальной и непротиворечивой. Поэтому организация процесса регулярной загрузки данных в Хранилище является важной задачей. Выделим основные этапы этого процесса: доступ к источникам данных; доставка на сервер репозитория Хранилища; преобразование данных (унификация, изменение структуры и т.д.); проверка на корректность и непротиворечивость, очистка; очистка; агрегирование.

Отличительной чертой системы SAS является тот факт, что данные доступные через дескрипторы SAS/ACCESS для всех процедур, работающих с данными, не отличаются друг от друга и от собственного формата хранения SAS. Это свойство системы позволяет легко изменять форматы внешних источников данных, практически не внося никаких изменений в Хранилище Данных. Единственным изменением будет смена дескриптора доступа к внешним данным.

Доставка на сервер. При реализации Хранилища Данных для организации, автоматизированные системы которой работают на нескольких серверах и даже на разных платформах, процесс транспортировки данных может создавать определенные проблемы. Система SAS функционирует на многочисленных платформах и имеет хорошо развитые средства межплатформенного общения:

SAS/CONNECT и SAS/SHARE между такими платформами как: MVS, VM/CMS, различные UNIX, OpenVMS VAX & AXP, OS/2, Window, Windows NT, Macintosh и другие.

Преобразование данных. Физическая структура Хранилища Данных часто сильно отличается от структуры источников. Основной причиной обычно является требования к эффективному исполнению запросов и прогнозируемое время отклика. Кроме изменения структуры, при интеграции данных из разных источников необходимо унифицировать форматы представления.

В системе SAS разработчику доступны следующие средства обработки и преобразования данных:

1. SQL – стандартный язык обработки реляционных данных;
2. Data Step – эффективный 4GL язык обработки данных, разработка SAS Institute;
3. SAS/IML – язык для работы с матрицами, в виде математической нотации;
4. Различные специализированные процедуры обработки данных на основе эффективных алгоритмов сортировки (SORT), преобразования временных рядов (EXPAND), шкалирования (RANK) и пр.

Проверка на корректность и очистка. Одним из самых важных свойств Хранилища Данных является достоверность доставляемой информации. Ричард Хекатрон, один из пионеров концепции, назвал Хранилище Данных как «единый образ истины» для всей организации. Поэтому, проверка на непротиворечивость и корректность загружаемых данных, а также очистка и снятие противоречий является ключевым элементом Хранилища Данных. Исследования, проведенные на работающих в промышленной эксплуатации Хранилищах Данных, показали, что 90% конечных пользователей нуждаются в сильно агрегированной информации, и лишь изредка возникает необходимость анализа информации детального уровня.

Общая структура репозитория Хранилища Данных является в своем роде отражением главной цели его построения, а именно, максимально полно и быстро удовлетворить потребности пользователей в той или иной информации. В зависимости от потребностей пользователей в информации можно выделить следующие основные типы: персональная информация; информация по бизнес темам; текущие детальные данные; старые детальные данные.

Персональная информация – это информация, используемая пользователями со строго определенными обязанностями и информационными потребностями. Обычно требует большой предварительной обработки или другими словами имеет высокий уровень агрегации (под агрегацией мы будем понимать не только суммирование, но и другие преобразования данных производимых с помощью как аддитивных так и не аддитивных статистик). Чаще всего хранятся в многомерных базах данных.

Информация по бизнес темам – информация, относящаяся к определенной тематике, такой, например, как финансовая деятельность организации. Для организаций имеющих, близкие функциональные и организационные структуры ее можно определить как информация для подразделения (например,

для финансовой службы). Имеет более широкий спектр, как в предметных областях, так и во времени, но в то же время напрямую используется реже, чем персонализированная информация. Обычно хранятся в смешанных структурах: многомерные базы данных и реляционные таблицы.

Текущие детальные данные – самая подробная информация доступная в Хранилище Данных. Обычными пользователями используется весьма редко, только в случае необходимости сильного уточнения информации. Обычно является полем деятельности аналитиков по поиску знаний (или поиску скрытых зависимостей в больших объемах информации). Обычно хранится в реляционных структурах.

Старые детальные данные – по сути, это тот же самый низкий уровень агрегирования, что и у текущих детальных данных. Выделяется в особый тип по следующей причине: с одной стороны, детальные данные часто требуют больших ресурсов для хранения, с другой стороны, детальные данные с возрастом, например, несколько лет, бывают необходимы в очень редких случаях. Решением в данном случае является использование более дешевых и емких способов хранения, например, ленты или роботизированные библиотеки. Для системы SAS доступ к таким данным не отличается от доступа к таблицам, хранящимся на диске. Единственное отличие – медленная скорость чтения, что делает реализацию такой схемы особенно легкой и предпочтительной.

Изучив данную тему, студент должен:

- *знать:* основные принципы организации процессов обработки данных.
- *уметь:* описать основные этапы технологии Data Mining и других.
- *иметь представление о:* процессах проектирования хранилищ данных.

При изучении темы необходимо:

- изучить учебный материал: [2, С. 54-61, 89-103]
- самостоятельно более подробно рассмотреть работы: 5, 6, 7.

Особое внимание обратить на: вопросы процесса проектирования хранилища данных.

Для самоконтроля по теме необходимо ответить на следующие вопросы.

1. Основная концепция построения систем с хранилищем данных.
2. Характеристика основных стадий построения Хранилища данных.
3. Какова цель этапа Оценка результатов?
4. Что такое метаданные Хранилища данных, и какую информацию они содержат?
5. Для кого и чего предназначены средства просмотра и поиска метаданных, и какую функциональность они имеют?

4. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ И ПОДГОТОВКА К МЕРОПРИЯТИЯМ ТЕКУЩЕГО КОНТРОЛЯ

В ходе проведения лекционных занятий предусматривается:

- использование мультимедийных презентаций по всем темам дисциплины;
- организация дискуссий по теме «Стандарты баз данных».
- творческая проблемно-ориентированная самостоятельная работа студентов включающая в себя поиск, анализ, структурирование и презентация информации по теме занятий (или индивидуальных заданий); анализ научных публикаций по заранее определенной преподавателем теме..

В ходе проведения всех лабораторных занятий предусматривается использование средств вычислительной техники при выполнении индивидуальных заданий и контрольной работы.

С целью формирования и развития профессиональных навыков обучающихся применяются интерактивные формы обучения на аудиторных занятиях. Учебный процесс, опирающийся на использование интерактивных методов обучения, организуется с учетом включенности в процесс познания всех студентов группы без исключения. Совместная деятельность означает, что каждый вносит свой особый индивидуальный вклад, в ходе работы идет обмен знаниями, идеями, способами деятельности.

Организуется индивидуальная, парная и групповая работа, используется проектная работа, ролевые игры, осуществляется работа с документами и различными источниками информации. Интерактивные методы основаны на принципах взаимодействия, активности обучаемых, опоре на групповой опыт, обязательной обратной связи. Создается среда образовательного общения, которая характеризуется открытостью, взаимодействием участников, равенством их аргументов, накоплением совместного знания, возможностью взаимной оценки и контроля.

Текущий, промежуточный и рубежный контроль проводится с помощью сервисов образовательного портала.

Исследовательские методы обучения-организация обучения на основе поисковой, познавательной деятельности студентов путем постановки преподавателем познавательных и практических задач, требующих самостоятельного творческого решения. Сущность исследовательского метода обучения обусловлена его функциями. Метод организует творческий поиск и применение знаний, является условием формирования интереса, потребности в творческой деятельности, в самообразовании. Основная идея исследовательского метода обучения заключается в использовании научного подхода к решению той или иной учебной задачи.

Работа студентов в этом случае строится по логике проведения классического научного исследования с использованием всех научно-исследовательских методов и приемов, характерных для деятельности ученых.

При подготовке к зачету особое внимание следует обратить на следующие моменты:

- выполнение практических заданий оказывает положительное влияние на усвоение теоретического материала;
- контрольные работы выполняются на примере реальных предметных областей, допускается самостоятельный выбор предметной области;
- при подготовке ответов по практическим заданиям следует всегда рассматривать примеры их применения;
- при проведении сравнительного анализа требуется знание методики его выполнения;
- при подготовке ответов следует активно использовать материалы контрольных работ и электронных презентаций.

Опыт приема зачета выявил, что наибольшие трудности при его проведении возникают во время ответов по следующим темам:

- концепция технологии хранилищ данных;
- администрирование систем с базами данных.

Для того чтобы избежать трудностей при ответах по вышеназванным темам, рекомендуем выполнить практические работы по разделам как для предметной области исследуемого процесса, так и для предметной области индивидуального домашнего задания; при изучении разделов дисциплины 4.1 и 4.2 изучить примеры, приводимые в [1, 2, 8, 9].

5. МАТЕРИАЛЫ ПО КОНТРОЛЮ ЗНАНИЙ

Вопросы для подготовки к зачету по дисциплине «Управление данными»

1. Файловый подход к организации информационной базы СОО – сущность подхода, достоинства и недостатки
2. Организация интегрированной информационной базы СОО – сущность подхода, достоинства и недостатки
3. Понятие СУБД, основные функции СУБД
4. Обеспечения безопасности и секретности данных
5. Избирательный подход к обеспечению безопасности данных
6. Обязательный подход к обеспечению безопасности данных
7. Контрольный след файла, модификация запроса как подходы к обеспечению безопасности данных
8. Безопасность в статистических БД
9. Проблемы обеспечения управляемой избыточности и целостности данных
10. Понятие транзакции, свойства транзакции, способы завершения транзакции
11. Основные подходы к обеспечению параллельного выполнения транзакций.
12. Проблемы параллельного выполнения транзакций
13. Проблема пропавших изменений
14. Проблема промежуточных данных
15. Проблема несогласованных данных
16. Язык SQL. Назначения, стандарты, подмножества языка.
17. Инфологическое моделирование: функциональный и предметный подходы к проектированию БД, проектирование с использованием метода «Сущность–связь»
18. Инфологическое моделирование. Модель «сущность–связь»: Сущности, классификация и характеристика сущностей
19. Инфологическое моделирование. Модель «сущность–связь»: Атрибуты, классификация и характеристика атрибутов
20. Инфологическое моделирование. Модель «сущность–связь»: Связи, классификация и характеристика связей
21. Инфологическое моделирование. Модель «сущность–связь»: Первичные и внешние ключи
22. Инфологическое моделирование. Модель «сущность–связь»: ограничения целостности.
23. Построение многомерной модели. Концепция, типы моделей, правила построения, область применения.

Задания для контрольной работы по теме «Технология хранилищ данных»

ВАРИАНТ 1

1. Дайте характеристику информации о некоем объекте с точки зрения решения конкретной задачи.
2. Сравните основные подходы к архитектуре Хранилищ данных.

3. Виртуальное хранилище данных.
4. Основы анализа данных: основные характеристики описательной статистики.
5. Размерная модель. Типы. Таблица факта.

ВАРИАНТ 2

1. Дайте характеристику процессу манипулирования данными: сбор, обработка, анализ, интеллектуальный анализ.
2. Основные подходы к архитектуре Хранилищ данных: хранилище данных с архитектурой шины Ральфа Кимболла.
3. Двухуровневая архитектура хранилища данных.
4. Основы анализа данных: суть корреляционного анализа.
5. Размерная модель. Типы. Таблица измерений.

ВАРИАНТ 3

1. Дайте характеристику концептуальной модели Хранилища данных.
2. Трехуровневая архитектура хранилища данных.
3. Основные подходы к архитектуре Хранилищ данных: корпоративная информационная фабрика Билла Инмона.
4. Основы анализа данных: суть регрессионного анализа.
5. Размерная модель. Типы. Консольная таблица.

6. ЗАДАНИЯ ДЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ

По дисциплине «Управление данными» предусмотрена аудиторная и внеаудиторная самостоятельная работа обучающихся.

Аудиторная самостоятельная работа студентов предполагает подготовку к выполнению заданий лабораторных работ.

Примерные темы лабораторных работ (содержание заданий представлено на образовательном портале по дисциплине «Управление данными»):

1. Предметная область банка данных;
2. Архитектура банка данных;
3. База данных как информационная модель предметной области;
4. Построение инфологической модели базы данных;
5. Обоснование выбора СУБД для реализации базы данных;
6. Построение многомерной модели базы данных хранилища данных

для киоска;

7. Структурированный язык запросов.

Внеаудиторная самостоятельная работа обучающихся осуществляется в виде изучения литературы по соответствующему разделу с проработкой материала; подготовки отчета в виде выполнения реферата.

Реферат - это краткое изложение в письменном виде содержания и результатов индивидуальной учебно-исследовательской деятельности, имеет регламентированную структуру, содержание и оформление.

Задачи реферата: формирование умений самостоятельной работы студентов с источниками литературы, их систематизация; развитие навыков логического мышления; углубление теоретических знаний по проблеме исследования.

Текст реферата должен содержать аргументированное изложение определенной темы. Реферат должен быть структурирован (по главам, разделам, параграфам) и включать разделы: введение, основная часть, заключение, список используемых источников. В зависимости от тематики реферата к нему могут быть оформлены приложения, содержащие документы, иллюстрации, таблицы, схемы и т.д.

Темы рефератов, рекомендуемые к написанию при изучении дисциплины «Управление данными».

1. Архитектура информационных систем.
2. Банки данных в информационных системах.
3. Централизованное управление данными с помощью СУБД.
4. Управление данными во внешней памяти.
5. Управление буферами оперативной памяти.
6. Промышленные СУБД. Распространенность и классификация.
7. Языки представления даталогических моделей.
8. Языки типа «query by example».
9. Объектно-ориентированные базы данных.
10. XML–базы данных.

11. RDF–базы данных.

Внеаудиторная самостоятельная работа обучающихся осуществляется в виде изучения литературы по соответствующему разделу с проработкой материала; выполнения комплексного индивидуального задания (КИЗ), по вариантам.

Индивидуальное задание:

1. Выполнить анализ предметной области и представить краткую постановку задачи на проектирование базы данных;
2. Выделить основные информационные потоки и функции обмена данными;
3. Построить функциональную модель бизнес процесса обработки информации;
4. Сформулировать типовые запросы, на которые система должна давать ответы, с обоснованием определения;
5. Построить концептуальную модель базы данных с использованием MySQL Workbench или любого другого средства.

Вариант 1

Спроектировать модель базы данных информационной системы согласно поставленной задаче. Учет наличия и движения товаров в торговой организации.

Модуль «Учет движения товаров на складе». В процессе учета участвуют специалисты следующих подразделений: склада, бухгалтерии, группы маркетинга, торгового зала. Товары подразделяются на товарные группы (бытовая техника, обувь, одежда, электроника и т.д.). Внутри группы товары отличаются наименованием, маркой, производителем, поставщиком и т.д.

Программное обеспечение кладовщика должно позволять:

- 1) хранить необходимую информацию о каждом виде товара, имеющегося на складе; хранить справочник нормативов запаса товаров по каждой группе товара;
- 2) автоматизировать обработку информации при следующих операциях:
 - прием товара от поставщиков (ввод данных приходной накладной);
 - выдача товара в торговый зал (ввод данных о расходе и оформление расходной накладной);
 - списание товара (ввод данных о списании и оформление акта о списании);
 - переоценка товара (ввод данных о новой цене заданного товара, групповое изменение цены с заданным коэффициентом);
 - передача устаревших документов в архив (накладные и акты за истекший финансовый год должны быть скопированы в архив и удалены из текущей БД).

Вариант 2

Спроектировать модель базы данных информационной системы согласно поставленной задаче. Учет основных средств (ОС) в автотранспортном предприятии.

Модуль «Учет ОС в подразделении предприятия». В процессе учета участвуют специалисты основных подразделений предприятия, бухгалтерии,

отдела материально-технического снабжения. ОС подразделяются на группы (здания, сооружения, станки, оборудование, автотранспорт грузовой, легковой и т.п.). Внутри группы ОС отличаются наименованием, маркой, производителем, каждое ОС имеет уникальный инвентарный номер.

Материально-ответственное лицо (МОЛ) в подразделении ведет инвентарные карточки ОС, ежегодно рассчитывает износ ОС, оформляет списание ОС. Учет ведется по мере движения ОС, в режиме реального времени. Инвентаризация проводится ежегодно совместно сотрудниками бухгалтерии и материально-ответственными лицами подразделений, по итогам составляются акты списания и переоценки ОС. Программное обеспечение материально-ответственного лица должно позволять:

1)хранить необходимую информацию о каждом ОС в форме инвентарной карточки;

2)автоматизировать обработку информации при следующих операциях:

- оформление заявки на ОС (ввод данных заявки);
- прием на учет нового ОС (ввод данных об ОС в инвентарной карте);
- списание ОС (ввод данных о списании, оформление акта о списании),
- переоценка ОС (изменение суммы износа в сведениях об ОС).

Вариант 3

Спроектировать модель базы данных информационной системы согласно поставленной задаче. Учет заказов на продукцию на малом предприятии.

Модуль «Работа с клиентами». Малое предприятие принимает от населения и предприятий заказы на изготовление продукции (например, мебели). В процессе учета участвует менеджер по работе с клиентами, бухгалтерия, менеджер по снабжению, управляющий производством. Менеджер по работе с клиентами оформляет заказы, рассчитывает стоимость заказа, принимает оплату и выдает готовую продукцию. На заказанную продукцию принимается предоплата и выдается квитанция и кассовый чек. В конце рабочего дня подсчитывается кассовая выручка, оформляются документы о сдаче денег. Постоянные клиенты пользуются скидкой в 5% от стоимости заказа.

Программное обеспечение регистратора должно позволять:

1)хранить в течение года необходимую информацию о каждом клиенте и сделанных им заказах; хранить преysкурaнт продукции;

2)автоматизировать обработку информации при следующих операциях:

–прием нового заказа (ввод новой записи в книге заказов, формирование квитанции -счета);

–коррекция принятого заказа (поиск заказа и изменение количества или видов заказанной продукции);

–отпуск готовой продукции (формирование счета на оплату с учетом скидки, отметка о выполнении заказа);

–предоставление клиентам рекламной информации о продукции фирмы (выпуск рекламных буклетов).

Вариант 4

Спроектировать модель базы данных информационной системы согласно поставленной задаче. Учет заказов на продукцию на малом предприятии.

Модуль «Управление производством». Малое предприятие принимает от населения и предприятий заказы на изготовление продукции (например, мебели). В процессе учета участвует менеджер по работе с клиентами, бухгалтерия, менеджер по снабжению, управляющий производством. Менеджер распределяет преподавателей -консультантов между группами студентов в соответствии с изучаемыми дисциплинами и учитывает количество оказанных образовательных услуг. В конце месяца ведомости оказанных образовательных услуг передаются в бухгалтерию (отдельно для преподавателей и для студентов).

Программное обеспечение менеджера должно позволять:

1)хранить сведения о преподавателях, справочник специализации каждого преподавателя (какие предметы он может вести), прейскурант образовательных услуг, в течение учебного года -график работы преподавателей;

2)автоматизировать обработку информации при следующих операциях:

–прием или увольнение преподавателя;

–составление и коррекция графика работы преподавателя;

–ввод данных об образовательных услугах с группировкой по предметам;

–изменение данных о цене образовательных услуг, например, повышение цены на услуги по заданному предмету на X руб.

–ежемесячное формирование ведомости оказанных образовательных услуг по каждому преподавателю.

Вариант 5

Спроектировать модель базы данных информационной системы согласно поставленной задаче. Автоматизация канцелярской деятельности учреждения.

Модуль «Внешняя корреспонденция». Канцелярия учреждения организует документооборот входящей, исходящей, внутренней организационно-распорядительной документации. К корреспонденции относятся письма, телеграммы, факсы, электронные письма. На каждый документ может быть заведена регистрационная карточка. В канцелярии поддерживается справочник подразделений учреждения и справочник организаций, с которыми ведется переписка. По указанию руководства периодически анализируется интенсивность документооборота (количество документов за определенный период или по определенному адресу). Один из сотрудников регистрирует внешнюю корреспонденцию в специальном журнале и передает ее по назначению.

Программное обеспечение этого сотрудника должно позволять:

1)хранить необходимую информацию о каждом виде внешней корреспонденции; хранить справочник подразделений предприятия и справочник внешних корреспондентов;

2)автоматизировать обработку информации при следующих операциях:

–регистрация корреспонденции (ввод данных об отправленных и пришедших письмах и пр.);

–ведение справочника внешних корреспондентов (ввод и коррекция данных);

–анализ интенсивности документооборота (формирование типовых диаграмм и отчетов).

Вариант 6

Спроектировать модель базы данных информационной системы согласно поставленной задаче. Жилищный кооператив.

Модуль «АРМ паспортистики». В кооперативе имеется несколько многоквартирных домов. Некоторые жильцы имеют льготы по оплате некоторых услуг. Коммунальные организации (Водоканал, Горгаз и т.п.) информируют кооператив о неплательщиках, эта информация отображается в ведомости оплаты коммунальных услуг.

Паспортистка кооператива ведет книгу учета жильцов, ведет список жильцов, имеющих льготы по оплате коммунальных услуг, выдает жильцам справки о составе семьи, о жилплощади и др.

Программное обеспечение паспортистики должно позволять:

1)хранить данные о жильцах, о льготниках и о квартирах;

2)автоматизировать обработку информации при следующих операциях:

–прописка нового жильца и текущая коррекция данных о жильцах в домовоей книге, отметка владельцев квартир;

–выписка жильца (данные о выписанном жильце должны быть скопированы в архив и удалены из текущей БД);

–обмен квартирами (жильцы одной квартиры переселяются в другую квартиру и наоборот);

–внесение сведений о льготах, предоставляемых конкретному жильцу по заданному виду коммунальных услуг;

–выдача справок жильцам.

Вариант 7

Спроектировать модель базы данных информационной системы согласно поставленной задаче. Мониторинг закупочных цен предприятия.

Модуль «АРМ маркетолога». Предприятие производит закупки материалов для изготовления своей продукции (например, колбасных изделий) у различных поставщиков. Ряд поставщиков принимает оплату за материалы в виде готовой продукции, другие -только «живыми деньгами». Задача мониторинга состоит в отслеживании рыночных цен на материалы и подборе наилучших поставщиков с учетом их географического расположения (в фактическую стоимость материала следует включить расходы на доставку). Мониторингом занимается отдел маркетинга, который аккумулирует информацию о поставщиках и их предложениях, ведет переписку с ними, заключает договора о намерениях, вырабатывает рекомендации по закупкам и т.д.

Программное обеспечение маркетолога должно позволять:

1) хранить данные о поставщиках материалов, о заключенных с ними договорах и сведения о транспортных расходах в зависимости от города, в котором расположен поставщик;

2)автоматизировать обработку информации при следующих операциях:

–заключение договора с новым поставщиком (ввод данных о поставщике и договоре);

–изменение данных о поставщике (адрес, ИНН, список поставляемых товаров, их цены);

–разрыв всех отношений с заданным поставщиком (удаление данных о нем и его договорах в архив);

–выбор наилучшего поставщика (с минимальной фактической ценой материала) для каждого вида материала.

Вариант 8

Спроектировать модель базы данных информационной системы согласно поставленной задаче. Отделение страховой компании.

Модуль «АРМ страхового агента». Страховая компания заключает договора страхования с физическими и юридическими лицами. В процессе заключения договора проверяется наличие предыдущих договоров, случаи страховых выплат, рассчитываются поправки к тарифной ставке страхования. Поэтому ИС должна длительное время хранить сведения о заключенных договорах и выдавать необходимые документы и справки. В случае наступления страхового события рассчитывается возмещение ущерба, пересчитывается или прекращается договор страховки. Каждый договор ведет выделенный сотрудник (агент), который получает комиссионные от суммы договора. Агент находит клиентов, заключает с ними договор страхования, переоформляет договор после окончания срока страхования, рассчитывает ущерб при наступлении страхового случая, составляет отчеты о своей работе и передает их в бухгалтерию.

Программное обеспечение страхового агента должно позволять:

1)хранить данные о клиентах, о заключенных с ними договорах, справочники для расчета суммы возмещения ущерба и страхового взноса в зависимости от вида и срока страхования;

2) автоматизировать обработку информации при следующих операциях:

–заключение договора с новым клиентом (ввод данных о клиенте и договоре, вывод печатной копии договора);

–расчет суммы возмещения ущерба;

–переоформление заданного договора (изменение данных о сумме страхового взноса);

–формирование ежемесячных отчетов о работе.

ГЛОССАРИЙ

Автоматизированная информационная система - совокупность программных и аппаратных средств, предназначенных для хранения и/или управления данными и информацией и производства вычислений; комплекс технических, программных, других средств и персонала, предназначенный для автоматизации различных процессов.

Архитектура информационной системы - концепция, определяющая модель, структуру, выполняемые функции и взаимосвязь компонентов информационной системы.

Защищенность информационной системы - способность системы противостоять несанкционированному доступу к конфиденциальной информации, ее искажению или разрушению. Различают два аспекта защищенности: техническую защиту (свойство недоступности) и социальную защищенность (свойство конфиденциальности).

Интерактивная информационная система - информационно-вычислительная система, в которой передача и обмен информацией происходят в режиме диалога.

Информационная безопасность - по законодательству РФ - состояние защищенности информационной среды общества, обеспечивающее ее формирование, использование и развитие в интересах граждан, организаций, государства.

Информационная безопасность имеет три основные составляющие:

- 1- конфиденциальность - защита чувствительной информации от несанкционированного доступа;
- 2- целостность - защита точности и полноты информации и программного обеспечения;
- 3- доступность - обеспечение доступности информации и основных услуг для пользователя в нужное для него время.

Информационная система - по законодательству РФ - организационно упорядоченная совокупность документов (массивов документов) и информационных технологий, в том числе с использованием средств вычислительной техники и связи, реализующих информационные процессы. Информационные системы предназначена для хранения, обработки, поиска, распространения, передачи и предоставления информации.

Информационная система - в модели OSI - совокупность, состоящая из одного либо нескольких компьютеров, соответствующих средств программирования, операторов, физических процессов, средств телекоммуникаций и других, образующих автономное целое, способное осуществлять обработку данных или передачу данных.

Информационная сеть - сеть, предназначенная для обработки, хранения и передачи данных. Информационная сеть состоит: из абонентских и административных систем, и из связывающей их коммуникационной сети. В зависимости от расстояния между абонентскими системами, информационные

сети подразделяются на глобальные, территориальные и локальные. Различают универсальные и специализированные информационные сети.

Информационно-поисковая система - система, выполняющая функции: хранения больших объемов информации; быстрого поиска требуемой информации; добавления, удаления и изменения хранимой информации; вывода информации в удобном для человека виде. Различают: автоматизированные (computerised); библиографические (reference); диалоговые (online); документальные и фактографические информационно-поисковые системы.

Информационное наполнение - тексты, графика, мультимедиа и иное информационно значимое наполнение информационной системы. Существенными параметрами информационного наполнения являются его объем, актуальность и релевантность. В сети Интернет информационное наполнение организуется в виде веб-страниц средствами языка гипертекстовой разметки.

Информационное пространство - совокупность информационных ресурсов банков и баз данных, средств информационного взаимодействия технологий их сопровождения и использования, информационная инфраструктура информационных телекоммуникационных систем, функционирующих на основе общих принципов и обеспечивающих информационное взаимодействие организаций и граждан, а также удовлетворение их информационных потребностей.

Основными компонентами информационного пространства являются: информационные ресурсы, средства информационного взаимодействия и информационная инфраструктура.

Информационный массив - совокупность зафиксированной информации, предназначенная для хранения и использования и рассматриваемая как единое целое. Информация может быть зафиксирована в виде публикаций, отчетов, электронных записей, микрокопий и т.д. Обычно на предприятиях и в учреждениях информационные массивы формируются по функциональному признаку.

Пользователь информационной системы - лицо, группа лиц или организация, пользующееся услугами информационной системы для получения информации или решения других задач.

Прокси-сервер - специальный Интернет-сервер, управляющий входящим и исходящим трафиком интернета в локальной сети. Прокси-сервер определяет безопасность передачи сообщений или файлов в сеть организации; управляет доступом к сети; фильтрует и отклоняет запросы согласно заданным параметрам.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Базы данных [Электронный ресурс]: учебное пособие / М.В. Махмутова; ФГБОУ ВПО «Магнитогорский государственный технический университет им. Г.И. Носова». – Электрон. текстовые дан. – Магнитогорск: ФГБОУ ВПО «МГТУ», 2015. – Систем. требования: Adobe Acrobat Reader. – Режим доступа: <http://magtu.ru:8085/marcweb2/Default.asp>. – Загл. с экрана
2. Балдин К.В., Уткин В.Б. Информационные системы в экономике. – М.: Дашков и К: 7-е изд. 2012. – 395 с. Рек. УМО. – Режим доступа: http://e.lanbook.com/books/element.php?pl1_id=3591 – Загл. с экрана – ISBN 978-5-394-01449-9
3. Бирюков А.А. Информационная безопасность: защита и нападение [Электронный ресурс] – М.: ДМК Пресс, 2012. – 474 с. – Режим доступа: http://e.lanbook.com/books/element.php?pl1_id=39990. – ISBN 978-5-94074-647-8
4. Введение в технологии баз данных [Электронный ресурс]: учебное пособие / М.В. Махмутова; ФГБОУ ВПО «Магнитогорский государственный технический университет им. Г.И. Носова». – Электрон. текстовые дан. – Магнитогорск: ФГБОУ ВПО «МГТУ», 2015. – Систем. требования: Adobe Acrobat Reader. – Режим доступа: <http://magtu.ru:8085/marcweb2/Default.asp>. – Загл. с экрана.
5. Интеллектуальные системы [Электронный ресурс]: учебное пособие / Е. А. Ильина, А. Ю. Миков, С. И. Файнштейн. МГТУ. - Магнитогорск: МГТУ, 2017. - 1 электрон. опт. диск (CD-ROM). - Режим доступа: <https://magtu.informsystema.ru/uploader/fileUpload?name=3396.pdf&show=dcatalogues/1/1139433/3396.pdf&view=true>. - Макрообъект. - ISBN 978-5-9967-103
6. Информационные технологии в профессиональной деятельности специалиста [Электронный ресурс]: учебное пособие / Демиденко Л.Л., Баранков В.В., Баранкова И.И.; ФГБОУ ВПО «Магнитогорский государственный технический университет им. Г.И. Носова». – Электрон. текстовые дан. – Магнитогорск : ФГБОУ ВПО «МГТУ», 2015. – Систем. требования : Adobe Acrobat Reader . – Режим доступа : <http://magtu.ru:8085/marcweb2/Default.asp>. – Загл. с экрана.
7. Рыбина Г. В. Основы построения интеллектуальных систем [Электронный ресурс] : учебное пособие для вузов / Г. В. Рыбина. – М. : Финансы и статистика, 2014. – 432 с. – Рек. УМО. - ISBN 978-5-279-03412-3. (1 экз) - <http://www.studentlibrary.ru/book/ISBN9785279034123.html>
8. Стружкин Н.П. Базы данных: проектирование: учебник для вузов/ Н.П.Стружкин, В.В.Годин. — Москва: Издательство Юрайт, 2020.— 477 с.— (Высшее образование).— ISBN 978-5-534-00229-4. — Текст : электронный // ЭБС Юрайт [сайт]. — URL:<https://urait.ru/bcode/450165> (дата обращения: 19.10.2020). Гриф УМО ВО
9. Стружкин Н.П. Базы данных: проектирование. Практикум: учебное пособие для вузов/ Н.П. Стружкин, В.В. Годин.— Москва: Издательство Юрайт, 2020.— 291 с.— (Высшее образование).— ISBN 978-5-534-00739-8. —

Текст : электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/451246> (дата обращения: 19.10.2020). Гриф УМО ВО

10. Теория экономических информационных систем [Электронный ресурс] : учебное пособие Изд.2-е, подгот.по печ.изд.2012 г./ О.Б. Назарова; ФГБОУ ВПО «Магнитогорский государственный технический университет им. Г.И. Носова». — Электрон. текстовые дан. — Магнитогорск: ФГБОУ ВПО «МГТУ», 2015. — Систем. требования : Adobe Acrobat Reader. — Режим доступа: <http://magtu.ru:8085/marcweb2/Default.asp>. — Загл. с экрана

11. Теория и практика прикладной информатики [Электронный ресурс] : хрестоматия / И. Д. Белоусова, И. В. Гаврилова, Л. З. Давлеткиреева и др. ; МГТУ. - Магнитогорск : МГТУ, 2016. - 1 электрон. опт. диск (CD-ROM). - Режим доступа:<https://magtu.informsystema.ru/uploader/fileUpload?name=2283.pdf&show=dcatalogues/1/1129893/2283.pdf&view=true>. - Макрообъект.

Программное обеспечение и Интернет-ресурсы

1. Официальные сайты аналитической информации и ИТ-компаний города:

- сайт по проектированию и разработке автоматизированных, информационных и аналитических систем. — Режим доступа: <http://www.info-system.ru>;
- сайт аналитической информации www.citforum.ru;
- www.aris-portal.ru;
- <http://www.ids-scheer.com>;
- www.sike.ru

2. *Программное обеспечение*: лицензионное программное обеспечение: операционная система MS Windows 2007; MS Office 2010; установленные на каждом персональном компьютере вычислительного центра ФГБОУ ВО «МГТУ».

Программное обеспечение: MS Access, MySQL, MsSQL Server

3. *Официальные сайты промышленных предприятий и организаций*:

- <http://www.mmk.ru>,
- <http://www.creditural.ru>,
- <http://www.magtu.ru>,
- <http://www.gks.ru> и т.п.;

разработчиков программных продуктов:

- <http://www.statsoft.ru>,
- <http://www.microsoft.com>,
- <http://www.ptc.com> и т.п.

4. Базы данных, информационно-справочные и поисковые системы:

— Единое окно доступа к образовательным ресурсам <http://window.edu.ru/>;

— Интернет-Университет Информационных Технологий <http://www.ict.edu.ru> <b,kbjntrf ext,ys[rehchd MSDN Academic Alliance <http://www.microsoft.com/Rus/Msdnaa/Curricula/Default.aspx>

ПРИЛОЖЕНИЯ

Приложение 1 Характеристика предметной области

Пример 1. Маркетинговая компания «GrounD Marketing» принимает от населения и предприятий заказы на продвижение продукции (например: товары, работы, услуги). В процессе учета участвует менеджер по работе с клиентами. Менеджер по работе с клиентами оформляет заказы. В конце рабочего дня печатается отчет по клиентам и заказам, которые находятся в работе. Программное обеспечение должно позволять:

1) хранить необходимую информацию о каждом клиенте и сделанных им заказах, об ответственном за выполнение заказа сотруднике компании;

2) автоматизировать обработку информации при следующих операциях:

– прием нового заказа (ввод новой записи в книге заказов);

Результаты анализа представлены в анкете.

Анкета:

1. Название подразделения: отдел по работе с клиентами.

1. Фамилия, имя, отчество начальника: Иванов Иван Иванович, телефон: (3519)224455.

2. Фамилия, имя, отчество заместителя начальника отдела: Петров Петр Петрович, телефон заместителя начальника отдела (3519)224456.

2. Основная цель отдела по работе с клиентами: поиск новых клиентов.

3. Основные функции отдела по работе с клиентами: поиск и работа с новыми клиентами.

4. Кадровый состав отдела по работе с клиентами: начальник отдела, заместитель начальника отдела. 3 сотрудника отдела, секретарь.

5. Документы, поступающие в отдел по работе с клиентами из других подразделений: данные о клиентах поступающие от других компаний.

6. Физическое представление, время и частота поступления, вид обработки и требования к безопасности каждого из этих документов: передача данных происходит в электронном виде (таблица), хранение на защищенном устройстве, использование для служебного пользования.

7. Продолжительность хранения каждого из документов и условия перевода в архив: отсутствуют.

8. Документы, формирующиеся в отделе по работе с клиентами отчет по клиентам которые находятся в работе.

9. Физическое представление время, частота и вид обработки и требования к безопасности каждого из этих документов: лист бумаги, электронный документ, каждый день, специальная папка.

10. Продолжительность хранения каждого из документов и условия перевода в архив: до момента прекращения общения с данными клиентами.

11. Документы, передающиеся в другие подразделения список согласившихся клиентов.

12. Физическое представление время и частота отправки, вид обработки и требования к безопасности каждого из этих документов: лист бумаги, электронный документ, каждый день, специальная папка.

13. Продолжительность хранения каждого из документов и условия перевода в архив: Срок хранения 1 год.

14. Описание информационной инфраструктуры (техническое оснащение и программные продукты) отдел по работе с клиентами: компьютерная периферия, МФУ.

15. Основные недостатки бизнес-процессов и существующей системы обработки документов в отделе по работе с клиентами: дублирование информации, качество связи.

16. Предложения по реорганизации бизнес-процессов и существующей системы обработки документов в отделе по работе с клиентами: ввод электронного документооборота, создание базы клиентов компании.

Таким образом, в результате проведенного анализа выделены: функции обработки информации: прием электронного документа, отправка электронного документа, запись в журнал, распечатка отчетов, и основные информационные объекты: таблица, отчет, журнал, БД.

Пример 2. ООО «PowerPC» занимается предоставлением услуг в сфере продаж компьютерной техники. Это продажа готовых компьютеров собственной сборки и комплектацией, имеет большой модельный ряд, а также занимается индивидуальной сборкой компьютеров под заказ.

Компания «PowerPC» начала свою работу в 2009 году, сотрудничает со многими именитыми компаниями как AMD, Intel, NVIDIA и другие.

Головной офис компании, шоурум и сборка находятся в городе Екатеринбурге. Компания осуществляет доставку по всей России и СНГ. А также сотрудничает как с частными лицами, так и с юридическими лицами.

Основные результаты обследования представлены в анкете.

Таким образом, в результате выделены функции обработки информации и основные информационные объекты.

- 1) Название подразделения: Отдел сборки и подготовки ПК к использованию.
- 2) Фамилия, имя, отчество начальника: Иванов И.И., телефон: +7-123-45-67-89
- 3) Фамилия, имя, отчество заместителя начальника отдела: Иванова И.И., телефон заместителя начальника отдела +7-123-54-45-45
- 4) Основная цель отдел сборки и подготовки ПК к использованию в: Подготовка компьютера для использования пользователем.
- 5) Основные функции отдел сборки и подготовки ПК к использованию в: Качественная сборка компьютера (Компьютеры модельного ряда/индивидуальные заказы), полная его настройка(вывод оптимальных настроек, разгон мощности комплектующих) и установка ПО (Операционная система, драйвера и программы по желанию заказчика), тестирование ПК (Обычно 8ч под нагрузкой/Среднее 24 Часов под нагрузкой/Полное 48 часов под нагрузкой)
- 6) Кадровый состав отдел сборки и подготовки ПК к использованию в: Управляющий отделом сборки, 3 сборщика компьютеров, 1 сборщик для эксклюзивных систем(Водяное охлаждение и т.п), винилограф, 2 настройщика и 2 QA-инженера.
- 7) Документы, поступающие в Отдел разработки IT Проектов из других подразделений: заказ клиента на сборку компьютера(ов) (Конфигурация(ии), перечень нужны программ), список комплектующих в наличии.
- 8) Продолжительность хранения каждого из документов и условия перевода в архив: Перевод документов в архив производится после истечения гарантии выполненной услуги.
- 9) Документы, формирующиеся в отделе разработки IT проектов: журнал учета выполненных заказов, журнал в ожидании, акт выполненных работ, журнал учет использованных комплектующих, журнал учета брака.
- 10) Продолжительность хранения каждого из документов и условия перевода в архив: Журналы хранятся до конца года, по истечению отправляется в архив. Прочие документы в архив не передаются. Акт выполненных работ передается в архив по истечению гарантии собранного компьютера. Журнал заказов в ожидании передается в архив после его полного заполнения.
- 11) Документы, передающиеся в другие подразделения: Акт выполненных работ, копия журнала учета выполненных заказов, журнал заказов в ожидании.
- 12) Физическое представление время и частота отправки, вид обработки и требования к безопасности каждого из этих документов: Отчеты – хранятся в электронном виде на ПК, часть журналов хранятся в электронном виде, часть в бумажном.

13) Продолжительность хранения каждого из документов и условия перевода в архив: Отчеты хранятся в течение трех лет и по истечении срока попадают в архив, журналы отправляют в архив ежегодно. Прочие документы в архив не передаются.

14) Описание информационной инфраструктуры (техническое оснащение и программные продукты) Отдел сборки и подготовки ПК к использованию: Компьютеры, периферия (Принтеры, МФУ), VR/AR оборудование, ОС: Windows, Linux ПО: Unity Engine, 3Ds MAX, Office Облачные системы: RedMine, хронометраж, сервер для хранения данных и архив.

15) Основные недостатки бизнес-процессов и существующей системы обработки документов в отдел сборки и подготовки ПК к использованию: сложность работы с документами, частичная различность в физическом представлении документов, дублирование, долгая передача части документов в другие подразделения, усложнен поиск работ в документах.

16) Предложения по реорганизации бизнес-процессов и существующей системы обработки документов в отделе разработки IT проектов: ввести электронный документооборот, основанный на автоматизированном учете документов в данной подразделении

Функции обработки информации: Ввод, вывод, сортировка, хранение

Основные информационные объекты: Предмет, процесс, текст, число, элемент, База Данных, таблица

Приложение 2

Построение внешней модели, функциональной модели обработки информации в нотации DFD

Пример 1. Для построения функциональной модели процесса обработки информации была выбрана нотация DFD.

Диаграмма потоков данных (data flow diagram, DFD) — один из основных инструментов структурного анализа и проектирования информационных систем. Далее будут представлены элементы нотации DFD, которые использовались для построения функциональной модели процесса обработки информации.

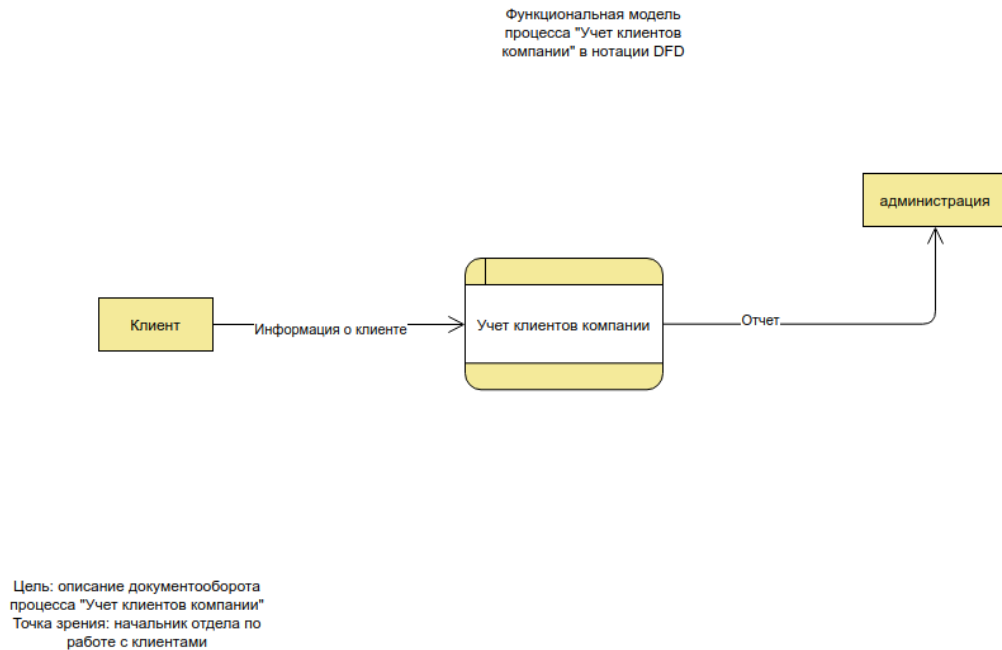
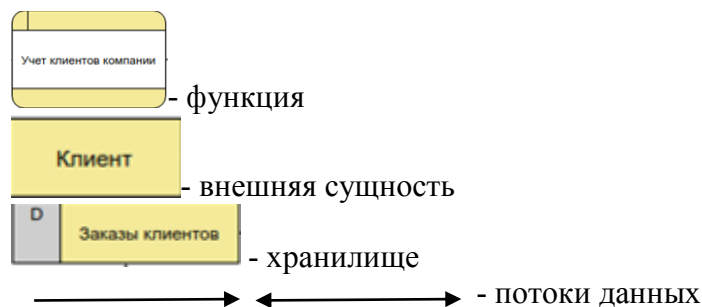


Рис. 2.1. Функциональная модель процесса обработки информации в нотации DFD.
Контекстная диаграмма

Документирование модели



Процессы:

Учет клиентов компании — основной процесс, в котором в БД вносятся сведения о клиентах компании, сведения о его заказе, сведения об ответственном лице и сроке исполнения заказа, стоимости.

Внешние сущности:

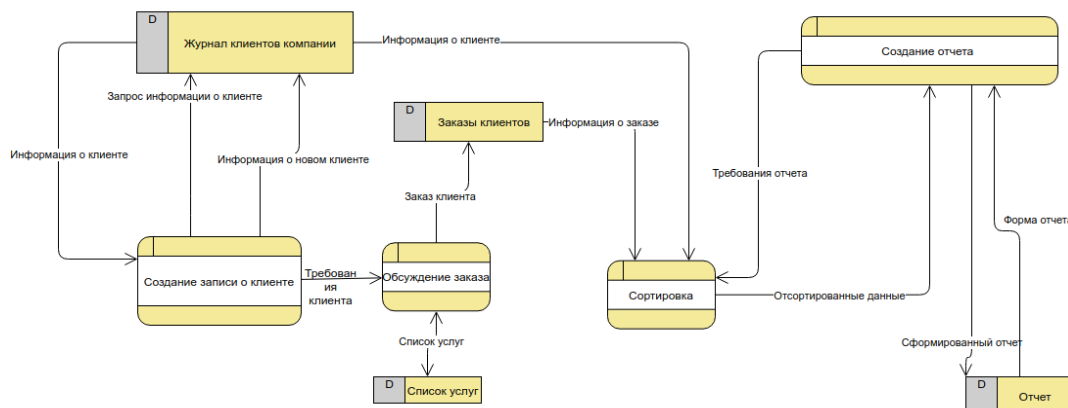
Клиент — ФЛ или ЮЛ.

Администрация — группа лиц, которая отвечает за анализ отчетов и работоспособность процесса «Учет клиентов компании».

Информационные потоки:

Информация о клиенте – ФЛ - данные (Серия и номер, ФИО), ЮЛ – наименование компании.

Информация о новом клиенте – ФЛ - данные (Серия и номер, ФИО), ЮЛ – наименование компании.



А0

Рис. 2.2. Функциональная модель процесса обработки информации в нотации DFD.
Главная диаграмма

Документирование модели

Процессы:

Создание записи о клиенте – процесс, в котором мы узнаем у клиента или в нашем журнале информацию нужную для создания записи в БД (паспортные данные, если ЮЛ добавляется название компании).

Обсуждение заказа – процесс, в котором клиент и представитель фирмы обсуждают заказ (определение срока выполнения, ответственного лица).

Информационные потоки:

Информация о клиенте – ФЛ - данные (Серия и номер, ФИО), ЮЛ – наименование компании.

Информация о новом клиенте – данные (Серия и номер, ФИО).

Запрос информации о клиенте – данные, которые нужны для создания записи о клиенте (паспортные данные, если ЮЛ добавляется название компании).

Заказ клиента – информация об ответственном лице, заказанной услуге, сроке исполнения, стоимости.

Список услуг – названия услуг и стоимость, которые мы можем оказывать по требованию заказчика.

Требования клиента – то с чем к нам приходит клиент (его пожелания по срокам выполнения, объему заказываемых услуг).

Информация о заказе – информация об ответственном лице, стоимости, услугах, которые должны быть выполнены в срок.

Отсортированные данные – данные, отсортированные под запрос для отчета.

Отчет о клиентах компании – документ, в котором находятся все клиенты, заказы которых выполняются в данный момент.

Хранилища данных:

Журнал клиентов компании – ФИО, серия и номер паспорта, ФЛ или ЮЛ, название компании.

Заказы клиентов – названия услуг, прописанные в договоре с клиентом.

Список услуг – Названия и стоимости оказываемых услуг.

Отчет формируется по запросу.

Пример 2. Для построения функциональной модели процесса обработки информации была выбрана нотация DFD.

DFD — Data Flow Diagram переводится как диаграмма потоков данных и используется для описания процессов верхнего уровня и для описания реально существующих в организации потоков данных.

На рисунке 1, будут представлены элементы, которые были использованы для построение модели поток данных.

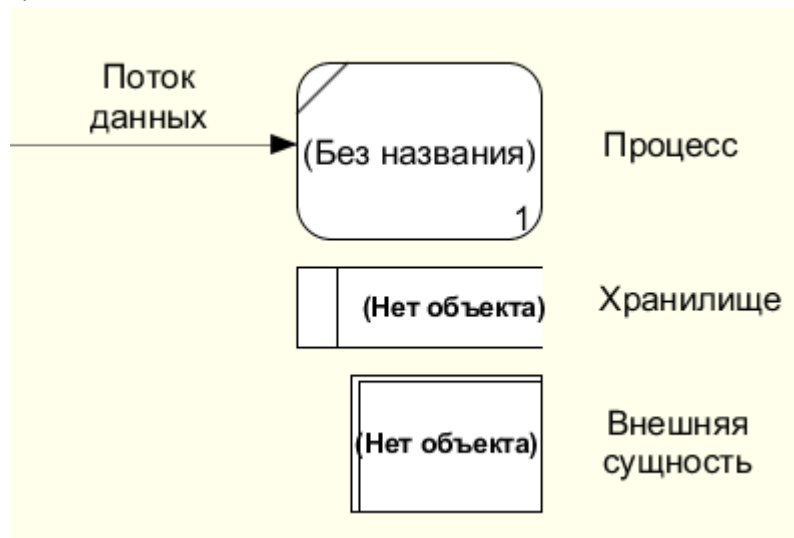


Рис. 2.3. Обозначения функциональных блоков для модели DFD

Модель DFD будет представлена на рисунках 2.4, 2.5.



Рис. 2.4. Функциональная модель процесса обработки информации в нотации DFD.
Контекстная диаграмма

Отчет по функциональной модели процесса обработки информации в нотации DFD.
Контекстная диаграмма

Процесс:

Сборка компьютера – основной процесс, в котором и происходит документооборот.

Внешние сущности:

Менеджер – сотрудник предприятия, которые работает с клиентом, и передает заказ в подразделение сборки ПК;

Склад комплектующих – Отдел предприятия, в котором храняться комплектующие для компьютеров;

Команда отдела сборки – Команда сотрудников, которые занимаются выполнением заказа.

Информационные потоки:

Запрос на наличие комплектующих – документ, в котором содержится список комплектующих для заказа;

Информация о наличии комплектующих – документ, в котором содержится список комплектующих в наличии;

Шаблон акта выполненных работ – пример составления акта выполненных работ;

Акт выполненных работ – документ, в котором содержится информация о выполненных работах над заказом;

Шаблон товарной накладной – пример составления товарной накладной;

Товарная накладная – документ, в котором содержится перечень использованных комплектующих;

Гарантия – документ, в котором обозначаются сроки службы ПК;

Шаблон гарантии – пример составления документа гарантии;

Информация о собранном компьютере – документ, в котором содержится информация о комплектации компьютера и произведенных работ над ним;

Информация о конфигурации компьютера – документ, в котором содержится информация о комплектующих использованных в сборке ПК, программное обеспечение;

ID компьютера – личный номер конфигурации компьютера выполненного заказа;

Информация о браке – документ, соборанную информацию о бракованных деталях обнаруженных во время сборки.

Информация о готовом заказе – документ, который содержит в себе номер заказа. Требования и номер комплектации компьютера, акт выполненных работ.

Отчет о браке – структурированный документ, со списком бракованных комплектующих;

Отчет о затратах на комплектующие – структурированный документ, в котором содержится информация о потраченных средствах на дополнительные комплектующие;

Запрос «дозаказанные комплектующие» – документ, который запрашивает информацию о дозаказанных комплектующих;

Форма отчета – составляется с соответствием запроса на отчет;

Информация о заказанных комплектующих – документ, содержащий в себе информацию о комплектующих, которых небыло в наличии.

Хранилища данных:

Заказы – хранилище, в котором содержится (ID заказа, ID клиента, требования, ID конфигурации, статус заказа, сроки заказа);

Комплектующие – хранилище, в котором содержится (ID Тип комплектующего, ID комплектующего, название комплектующего, характеристики);

Конфигурация ПК – хранилище, в котором содержится (ID конфигурации, Тип комплектующих, комплектующие, программное обеспечение);

Акт выполненных работ – документ, в котором содержится (ID акта выполненных работ, перечень выполненных работ, сотрудника, ID заказа, цена работ);

Товарная накладная – документ, в котором содержится (ID товарной накладной, тип комплектующего, комплектующее, цена комплектующего);

Гарантия – документ, в котором содержится (ID Гарантии, ID заказа, сроки гарантии);

Отчеты – документ, сформированные по соответствию с запросами;

Приложение 3

Определение информационных объектов предметной области

Пример 1.

Основные допущения относительно системы.

1. Клиент – юридическое лицо или ФЛ
2. Идентификационный номер клиента является уникальным
3. В договоре может быть одно или более наименований услуги
4. Идентификационный номер ответственного лица уникален
5. Код услуги – 6 числовых знаков
6. Нумерация документов начинается с нового года

На основе анализа функциональной модели процесса обработки информации, в нотации DFD, определим, какие элементы данных используются в каждом документе. Заполняем таблицу соответствия

Таблица 3.1

Соответствие информационных объектов DFD модели

	Элементы данных	Список услуг	Отчет	Журнал клиентов компании	Заказы клиентов	Договор
	НАЗВ-ДОК	+	+	+	+	+
	НОМ-ДОК		+		+	+
	ДАТА-СОСТАВЛЕНИЯ	+	+		+	+
	Ответственное лицо		+	+	+	+
	Клиент		+		+	+
	КОД-услуги	+	+		+	+
	НАИМЕНОВАНИЕ-услуги	+	+		+	+
	ЦЕНА	+	+		+	+
0	ПОДПИСЬ ответственного	+	+		+	
1	СУММА-ИТОГО		+		+	+
2	Серия и номер паспорта(ФЛ)			+		
3	Название компании (ЮЛ)			+		
4	ФИО			+		
5	Срок выполнения					+

Определим, какие взаимосвязи существуют между элементами и выделим информационные объекты.

Список услуг

Код услуги(РК)

Наименование услуги

Цена

Отчет

ID- отчета(РК)

ID-договора (FK)

Название
 Дата составления
 ID- ответственного лица (FK)
Клиент
 ID- клиента (PK)
 Фамилия
 Имя
 Отчество
 Название организации(ЮЛ)
 Серия и номер паспорта (ФЛ)
 Ответственное лицо (сотрудник)
 ID- ответственного лица (PK)
 Фамилия
 Имя
 Отчество
 Должность
Заказ
 ID- заказа (PK)
 ID- клиента (FK)
 Код услуги (FK)
 Сумма итого
 Дата заявки
Договор
 ID-договора (PK)
 ID- заказа (FK)
 ID- ответственного лица (FK)
 Дата подписания
 Сумма итого
 Сроки выполнения

Таким образом, были определены основные объекты БД. На основе этих объектов будет создана концептуальная модель БД.

Пример 2. Предметная область: сборка компьютера (Выполнение заказа). Определим основные допущения относительно системы.

1. Сотрудники компании – люди, работающие над заказом;
2. Идентификационный номер сотрудника является уникальным;
3. Идентификационный номер кладовщика уникален;
4. Идентификационный номер заказа уникален;
5. Идентификационный номер комплектующего уникален;
6. Код комплектующих – 5 числовых знаков + 1 буква пример (00001A, 99999X);
7. ID акта о выполненных работах уникален, и привязывается к заказу;
8. Гарантия начинается со дня составления «Акт выполненных работ», в данной схеме не учитывается;
9. Товарная накладная в данной схеме не учитывается;
10. Нумерация документов начинается заново с нового года.

Определим, какие элементы данных используются в каждом документе. Заполняем таблицу соответствия.

Определим, какие взаимосвязи существуют между элементами.

Комплектующие

ID комплектующего (PK);
 Тип комплектующего (FK);
 Название комплектующего;

Характеристики;
Кол-во;
Статус;
Цена.

Таблица 3.2

Соответствие информационных объектов DFD модели и их характеристик

	Элементы данных	Товарная накладная	Акт выполненных работ	Журнал учета заказов	Журнал учета конфигураций ПК	Гарантия	Журнал Комплектующие	Отчет о выполненном заказе	Отчет о затратах на комплектующие	Отчет о браке
1	НАЗВ-ДОК	+	+	+	+	+	+	+	+	+
2	НОМ-ДОК (ID)	+	+	+	+	+	+	+	+	+
3	ДАТА-СОСТАВЛЕНИЯ	+	+	+	+	+		+	+	+
4	КЛИЕНТ	+	+	+		+				
5	СОТРУДНИК	+	+	+		+		+	+	+
6	КОД-ЗАКАЗА	+	+	+	+	+				
7	КОД-КОМПЛЕКТУЮЩИХ	+	+	+	+	+	+		+	+
8	ПЕРЕЧЕНЬ РАБОТ		+					+		
9	КОЛИЧЕСТВО	+	+	+		+	+		+	+
10	ПОДПИСЬ ОТВЕТСТВЕННОГО	+	+	+		+		+	+	
11	ФИО	+		+		+		+	+	+
12	СРОКИ ВЫПОЛНЕНИЯ			+						
13	СРОК ДЕЙСТВИЯ					+				
14	ЦЕНА	+	+	+	+		+		+	+

Тип комплектующего

ID Тип комплектующего (PK);

Название типа.

Клиент

ID клиента (PK);

ФИО;

E-mail;

Телефон.

Сотрудник

ID сотрудника (РК);

ФИО;

E-mail;

Телефон.

Акт выполненных работ

ID акта выполненных работ (РК);

ID Вид работы;

ID Сотрудник;

Кол-во;

Цена.

Вид работ

ID вида работ (РК);

Название вида.

Комплектация

ID комплектация;

ID комплектующего;

Цена.

Заказ

ID заказа (РК);

ID Комплектация;

ID Клиент;

ID Сотрудник;

ID Акт выполненных работ;

Статус заказа;

Сроки;

Цена.

Приложение 4

Построение концептуальной схемы базы данных

Пример 1. Для построения онлайн концептуальной модели использовался онлайн-сервис Creately.

Creately — это простой в использовании диаграмма и программное обеспечение flowchart, построенное для совместной работы. Поддерживает более 40 типов диаграмм и имеет 1000 профессионально нарисованных шаблонов.

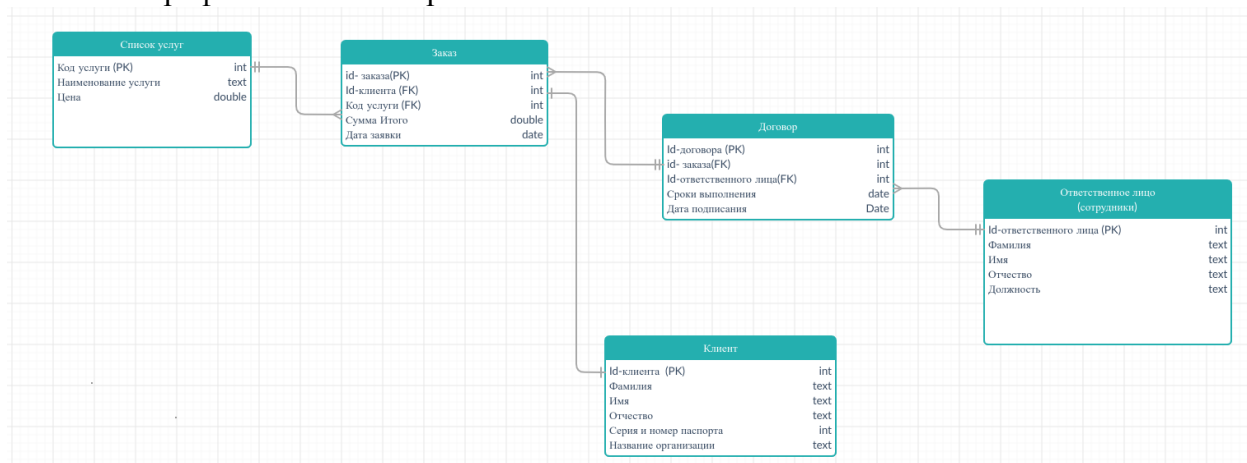


Рис. 4.1. Концептуальная модель БД

После построения концептуальной модели БД можем приступить к созданию реляционной БД.

Пример 2. Концептуальная модель базы данных для компании «PowerPC», представлена на рисунке 4.2.

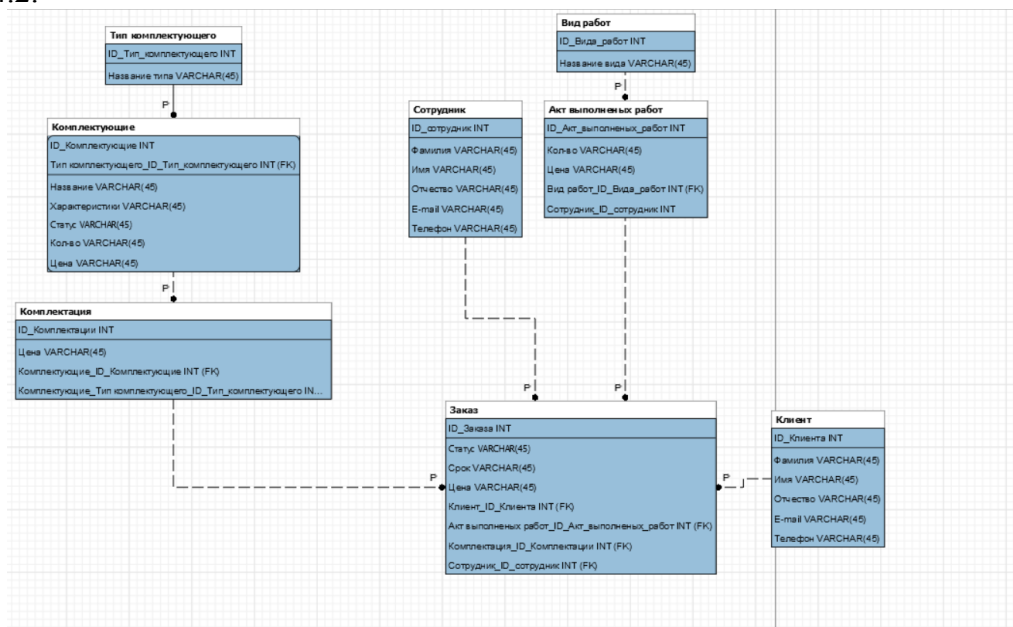


Рис. 4.2. Концептуальная модель данных для компании «PowerPC»

Приложение 5

Обоснование выбора СУБД для реализации базы данных

Следует отметить, что по существующей практике решение об использовании той или иной СУБД принимает один человек - обычно, руководитель предприятия, а он может опираться отнюдь не на технические критерии. Здесь свою роль могут сыграть такие, с технической точки зрения, незначительные факторы как рекламная раскрутка компании-производителя СУБД, использование конкретных систем на других предприятиях, стоимость. При этом последний фактор может трактоваться в двух противоположных смыслах в зависимости от финансового состояния и политики предприятия. С одной стороны, это может быть принцип, - чем дороже, тем лучше. С другой стороны - культивирование почти бесплатного использования продукта, вплоть до "взлома" его лицензионной защиты. Очевидно, последний подход чреват коллизиями и не может привести к успеху в долгосрочной работе.

Группы критериев, определяющих выбор СУБД

1. Моделирование данных.
2. Особенности архитектуры и функциональные возможности.
3. Контроль работы системы.
4. Особенности разработки приложений.
5. Производительность.
6. Требования к рабочей среде
7. Надежность.
8. Смешанные критерии.

Каждый из этих критериев может быть охарактеризован через набор определенных параметров. Рассмотрим каждую из выделенных групп критериев в отдельности для обоснования выбора СУБД LibreOffice Base, Microsoft SQL Server, PostgreSQL, MySQL и OpenOffice Base.

Четкий и глубокий сравнительный анализ на основании перечисленных критериев в любом случае поможет рационально выбрать подходящую систему для конкретного проекта, и затраченные усилия не будут напрасными. Перечень критериев поможет осознать масштабность задачи и выполнить ее адекватную постановку.

Характеристики параметров группы критериев, определяющих выбор СУБД LibreOffice Base, представлен в таблице 5.1.

Таблица 5.1

Группы критериев, определяющих выбор СУБД LibreOffice Base

Критерий	Параметр
1. Моделирование данных	<i>Используемая модель данных</i> Таблица состоит из строк и столбцов с данными. Если таблица содержит первичный ключ, то мы можем выбрать заданную строку, задав при поиске значение первичного ключа для этой строки. Каждый столбец таблицы содержит значения в конкретном поле. Поэтому, когда мы одновременно указываем значение первичного ключа и имя поля, мы можем выбрать конкретную ячейку таблицы. Это то, что нам нужно, если мы хотим найти конкретную информацию, которая содержится в таблице: путь с указанием строки и столбца для нашего поиска (адресация). LibreOffice Base может создавать и работать с двумя типами баз данных: плоскими и реляционными. Мы рассматриваем плоские базы данных. Реляционные базы данных подробно

	описаны в Base Handbook и в Руководстве по Base.
	<i>Предусмотренные типы данных</i> В LibreOffice Basic предусмотрена поддержка семи классов переменных. Логические переменные содержат одно из значений: TRUE или FALSE. Числовые переменные содержат числовые значения. Они могут быть целочисленные, целочисленные положительные, с плавающей точкой и с фиксированной точкой. Строковые переменные содержат строки символов. Переменные даты могут содержать дату и/или время во внутреннем формате. Объектные переменные могут содержать объекты различных типов. Специальные типы, такие как структуры, массивы. Абстрактный тип Variant.
	<i>Реализация языка запросов</i> В LibreOffice Base создавать запросы можно в режиме дизайна запросов, мастера запросов и SQL.
2. Особенности архитектуры и функциональные возможности	<i>Мобильность</i> Программа полностью переведена на русский язык и доступна для Linux, Windows и macOS.
	<i>Масштабируемость</i> С помощью Base создаются и эксплуатируются персональные базы данных с относительно небольшим объемом информации. Для создания крупных промышленных информационных систем данный пакет не подходит.
	<i>Сетевые возможности</i> LibreOffice Base позволяет создавать как локальные, так и сетевые реляционные базы данных или подключаться к уже имеющимся базам.
3. Контроль работы системы	<i>Контроль использования памяти компьютера</i> Память настраивается потребление программой LibreOffice оперативной памяти компьютера. На компьютерах с небольшим объемом оперативной памяти целесообразно отключать функцию Быстрого запуска, для ускорения работы компьютера.
4. Особенности разработки приложений	<i>Средства проектирования</i> СУБД LibreOffice Base позволяет пользователю, не прибегая к программированию, легко выполнять основные действия с базой данных: создание, редактирование и манипулирование данными.
	<i>Многоязыковая поддержка</i> Широкая поддержка языков. Пользовательский интерфейс LibreOffice доступен на более чем 40 языках. Проект

	LibreOffice предоставляет словари, схемы переносов и тезаурусы для более чем 70 языков и диалектов <i>Поддерживаемые языки программирования</i> Язык программирования LibreOffice очень гибок и позволяет автоматизировать как простые, так и сложные задачи. Макросы часто используются, когда мы хотим повторять ещё и ещё раз определённую последовательность действий в схожих условиях. Макросы в LibreOffice обычно записываются на языке, называемом LibreOffice Basic или просто Basic. Java, C++
5. Требования к рабочей среде	<i>Минимальные требования к оборудованию</i> Microsoft Windows 2000 (Service Pack 4 или выше), XP, Vista, Windows 7, или Windows 8; Pentium-совместимый процессор (Pentium III, Athlon или более современные); 256 Мб ОЗУ (512 Мб рекомендуемая); До 1.5 Гб свободного пространства на жестком диске; Разрешение 1024x768 (рекомендуется высокое разрешение), как минимум 256 цветов; Java SE Runtime Environment 7.0;
6. Надежность	<i>Откат изменений</i> LibreOffice функция контроля версий. Располагается она во вкладке «Файл → Версии». Контроль версий позволяет сохранить состояние документа в определённый момент времени и отслеживать все последующие изменения в нём. При необходимости, можно отменить внесённые изменения. Собственно, функция избавляет от ручного создания множества резервных файлов.

В LibreOffice Basic вся информация хранится в одном файле. Позволяет вводить данные в таблицы, контролировать правильность вводимых данных. Также можно создавать связи между таблицами, позволяющие использовать вместе данные из разных таблиц.

MS SQL Server – это платформа для решения критически важных задач в масштабе предприятия, обладающая высокой доступностью, повышенной производительностью и безопасностью. Решение представляет собой хорошо масштабируемый, полностью реляционный, быстродействующий сервер, способный обрабатывать большие объемы данных для клиент-серверных приложений. В конце 2010 года (то есть до выхода SQL Server 2012) вице-президент подразделения Microsoft Business Platform Division Тед Каммерт в интервью рассказал о планах по дальнейшему развитию продукта (как версии SQL Server 2012, так и будущих версий). В частности, Каммерт рассказал о том, что работа над SQL Server идёт в контексте идей Information Platform Vision, представляющей собой набор разнообразных возможностей, который ложится в основу платформы. SQL Server будет по-прежнему представлять собой единый продукт, реализуемый в настольных системах, в центрах обработки данных и в «облаке» (как в 32-битном, так и 64-битном варианте). Одним из приоритетных направлений по-прежнему останется бизнес-аналитика (англ. business intelligence, BI). С точки зрения Microsoft приоритетом в области бизнес-аналитики останется разработка средств BI, реализующих принцип самообслуживания, а также развитие экосистемы «облачных» вычислений.

SQL Server упрощает развертывание, передачу и интеграцию больших данных. Решение для обработки больших данных на основе Kubernetes, встроенное в SQL Server, позволяет легко развернуть кластер больших данных и работать с ними.

В SQL Server внешняя таблица или внешний источник данных обеспечивает соединение с Hadoop, виртуализируя внешние источники данных без необходимости их прямого импорта в реляционную базу, и потом позволяет обращаться к этим данным с запросами.

Защита конфиденциальных данных при помощи технологии Always Encrypted с защищенными анклавами. Шифрование на месте позволяет выполнять криптографические операции с конфиденциальными данными без их перемещения за пределы базы данных.

Характеристики параметров группы критериев, определяющих выбор СУБД MS SQL Server, представлен в таблице 5.2.

Таблица 5.2

Группы критериев, определяющих выбор СУБД MS SQL Server

<i>Критерий</i>	<i>Параметр</i>
1. <i>Моделирование данных</i>	Это платформа для решения критически важных задач в масштабе предприятия, обладающая высокой доступностью, повышенной производительностью и безопасностью. Решение представляет собой хорошо масштабируемый, полностью реляционный, быстродействующий сервер, способный обрабатывать большие объемы данных для клиент-серверных приложений. SQL Server учитывает все современные требования по работе с данными различных форматов и из разнообразных источников и становится естественным выбором для построения платформы интеграции, управления и анализа любых данных.
	<i>Интеграция структурированных и неструктурированных данных</i> В SQL Server внешняя таблица или внешний источник данных обеспечивает соединение с Hadoop, виртуализируя внешние источники данных без необходимости их прямого импорта в реляционную базу, и потом позволяет обращаться к этим данным с запросами.
2. <i>Особенности архитектуры и функциональные возможности</i>	Современные разработки в контексте идей Information Platform Vision, представляющей собой набор разнообразных возможностей, который ложится в основу платформы. Одним из приоритетных направлений по-прежнему останется бизнес-аналитика (англ. business intelligence, BI). С точки зрения Microsoft приоритетом в области бизнес-аналитики останется разработка средств BI, реализующих принцип самообслуживания, а также развитие экосистемы «облачных» вычислений.
3. <i>Контроль работы системы</i>	SQL Server представляет собой единый продукт, реализуемый в настольных системах, в центрах обработки данных и в «облаке» (как в 32-битном, так и 64-битном варианте).

4. Особенности разработки приложений	<i>Интеллектуальный анализ данных</i> Развитие SQL Server пошло по пути интеграции с другими аналитическими платформами, в частности Spark, которая включена теперь в поставку SQL Server. Spark является очень популярным инструментом для машинного обучения, для продвинутой аналитики, имеет эффективную in memory машину. И все это интегрировано с SQL, который очень эффективен для визуализации аналитики.
5. Производительность	Рекордная производительность MS SQL Server обеспечивается новыми технологиями работы с памятью, что помогает предприятиям ускорить свой бизнес и реализовать новые сценарии работы. Кроме того, SQL Server позволяет использовать новые гибридные облачные решения и пользоваться новыми преимуществами облачных вычислений. Расширенные функции безопасности, в сочетании со встроенными, удобными для использования инструментами и управляемым доступом к данным, позволяют организации выполнить требования строгих политик соответствия нормам.
6. Надежность	<i>Безопасность и соответствие требованиям</i> Защита конфиденциальных данных при помощи технологии Always Encrypted с защищенными анклавами. Шифрование на месте позволяет выполнять криптографические операции с конфиденциальными данными без их перемещения за пределы базы данных.

PostgreSQL - это мощная система объектно-реляционных баз данных с открытым исходным кодом, которая использует и расширяет язык SQL в сочетании со многими функциями, которые безопасно хранят и масштабируют самые сложные рабочие нагрузки данных.

PostgreSQL является бесплатным и открытым исходным кодом, он обладает широкими возможностями расширения. Например, вы можете определять свои собственные типы данных, создавать собственные функции, даже писать код на разных языках программирования без перекомпиляции базы данных.

PostgreSQL создана на основе некоммерческой СУБД Postgres, разработанной как open-source проект в Калифорнийском университете в Беркли. К разработке Postgres, начавшейся в 1986 году, имел непосредственное отношение Майкл Стоунбрейкер. Название расшифровывалось как «Post Ingres», и при создании Postgres были применены многие ранние наработки.

Стоунбрейкер и его студенты разрабатывали новую СУБД в течение восьми лет с 1986 по 1994 год. В 1995 году разработка снова разделилась: Стоунбрейкер использовал полученный опыт в создании коммерческой СУБД Illustra, продвигаемой его собственной одноименной компанией его студенты разработали новую версию Postgres - Postgres95, в которой язык запросов POSTQUEL - наследие Ingres - был заменен на SQL.

Разработка Postgres95 была выведена за пределы университета и передана команде энтузиастов. Новая СУБД получила имя, под которым она известна и развивается в текущий момент - PostgreSQL.

Основные возможности PostgreSQL

Функции являются блоками кода, исполняемыми на сервере. Хотя они могут писаться на чистом SQL, реализация дополнительной логики выходит за рамки SQL и требует

использования некоторых языковых расширений. Функции могут писаться с использованием одного из следующих языков:

- Встроенный процедурный язык PL/pgSQL, во многом аналогичный языку PL/SQL, используемому в СУБД Oracle;
- Скриптовые языки — PL/Lua, PL/LOLCODE, PL/Perl, PL/PHP, PL/Python, PL/Ruby, PL/sh, PL/Tcl, PL/Scheme, PL/v8 (Javascript);
- Классические языки — C, C++, Java (через модуль PL/Java);
- Статистический язык R (через модуль PL/R).

Архитектура PostgreSQL. PostgreSQL может применяться в среде клиент-сервер, что дает массу преимуществ как пользователям, так и разработчикам.

Основа PostgreSQL составляет серверный процесс БД. Он выполняется на одном сервере. Доступ из приложений к данным базы осуществляется посредством процесса БД. Клиентские программы не могут получить доступ к данным самостоятельно, даже если они работают на том же компьютере, на котором выполняется серверный процесс.

Такое разделение клиентов и сервера позволяет построить распределенную систему. Можно отделить клиентов от сервера посредством сети и разрабатывать клиентские приложения в среде, удобной для пользователя. Например, можно реализовать базу данных под UNIX и создать клиентские приложения, которые будут работать в системе Microsoft Windows.

Несколько клиентов подсоединяются к серверу по сети. PostgreSQL ориентирован на протокол TCP/IP - это может быть локальная сеть или Интернет. Каждый клиент соединяется с основным серверным процессом базы данных, который создает новый серверный процесс специально для обслуживания запросов на доступ к данным конкретного клиента.

Благодаря тому, что манипулирование данными сосредоточено на сервере, СУБД не приходится контролировать многочисленных клиентов, получающих доступ в совместно используемый каталог сервера, и PostgreSQL может поддерживать целостность данных даже при одновременном доступе большого количества пользователей.

Клиентские приложения соединяются с базой по специальному протоколу PostgreSQL. Однако можно установить на стороне клиента программное обеспечение, предоставляющее стандартный интерфейс для работы с нужным приложением, например, по стандарту ODBC или JDBC. Доступность ODBC-драйвера позволяет применять PostgreSQL в качестве базы данных для многих существующих приложений, включая такие продукты Microsoft Office, как Excel и Access.

PostgreSQL поддерживает большой набор встроенных типов данных:

- 1) Численные типы
 - Целые
 - С фиксированной точкой
 - С плавающей точкой
 - Денежный тип (отличается специальным форматом вывода, а в остальном аналогичен числам с фиксированной точкой с двумя знаками после запятой)
- 2) Символьные типы произвольной длины
- 3) Двоичные типы (включая BLOB)
- 4) Типы «дата/время» (полностью поддерживающие различные форматы, точность, форматы вывода, включая последние изменения в часовых поясах)
- 5) Булев тип
- 6) Перечисление
- 7) Геометрические примитивы
- 8) Сетевые типы
 - IP и IPv6-адреса
 - CIDR-формат
 - MAC-адрес
- 9) UUID-идентификатор

- 10) XML-данные
- 11) Массивы
- 12) JSON
- 13) Идентификаторы объектов БД
- 14) Псевдотипы

Более того, пользователь может самостоятельно создавать новые требуемые ему типы и программировать для них механизмы индексирования с помощью GiST.

Пользовательские объекты. PostgreSQL может быть расширен пользователем для собственных нужд практически в любом аспекте. Есть возможность добавлять собственные:

- Преобразования типов
- Типы данных
- Домены (пользовательские типы с изначально наложенными ограничениями)
- Функции (включая агрегатные)
- Индексы
- Операторы (включая переопределение уже существующих)
- Процедурные языки

Безопасность. Аутентификация: GSSAPI, SSPI, LDAP, SCRAM-SHA-256, сертификат и многое другое

- Надежная система контроля доступа
- Безопасность на уровне столбцов и строк
- Многофакторная аутентификация с сертификатами и дополнительным методом

Прочие возможности

- Соблюдение принципов ACID
- Соответствие стандартам ANSI SQL-92, SQL-99, SQL:2003, SQL:2011
- Поддержка запросов с OUTER JOIN, UNION, UNION ALL, EXCEPT, INTERSECT и

подзапросов

- Последовательности
- Контроль целостности
- Репликация
- Общие табличные выражения и рекурсивные запросы
- Аналитические функции
- Поддержка Юникода (UTF-8)
- Поддержка регулярных выражений в стиле Perl
- Встроенная поддержка SSL, SELinux и Kerberos
- Протокол разделяемых блокировок
- Подгружаемые расширения, поддерживающие SHA1, MD5, XML
- Расширения для написания сложных выборов, отчётов и т. д. (API открыт)
- Средства для генерации совместимого с другими системами SQL-кода и импорта из

других систем

- Автономные блоки на доступных языках, а не только SQL

Качество исходного кода. Согласно результатам автоматизированного исследования, различного ПО на предмет ошибок, проведённом в 2005 году, в исходном коде PostgreSQL было найдено 20 проблемных мест на 775 000 строк исходного кода (в среднем, одна ошибка на 39 000 строк кода). Для сравнения: MySQL - 97 проблем, одна ошибка на 8 000 строк кода; FreeBSD (целиком) - 306 проблем, одна ошибка на 2 500 строк кода; Linux (только ядро) - 950 проблем, одна ошибка на 800 строк кода.

Достоинства PostgreSQL. Открытое ПО соответствующее стандарту SQL - PostgreSQL - бесплатное ПО с открытым исходным кодом. Эта СУБД является очень мощной системой. Большое количество дополнений - несмотря на огромное количество встроенных функций, существует очень много дополнений, позволяющих разрабатывать данные для этой СУБД и управлять ими. Расширения - существует возможность расширения функционала за счет

сохранения своих процедур. Объектность – PostgreSQL это не только реляционная СУБД, но также и объектно-ориентированная с поддержкой наследования и много другого

Недостатки PostgreSQL. Производительность - при простых операциях чтения PostgreSQL может значительно замедлить сервер и быть медленнее своих конкурентов, таких как MySQL.

MySQL - это быстрая, надежная, открыто распространяемая СУБД. MySQL, как и многие другие СУБД, функционирует по модели "клиент/сервер". Под этим подразумевается сетевая архитектура, в которой компьютеры играют роли клиентов либо серверов. На рис. изображена схема передачи информации между компьютером клиента и жестким диском сервера.

MySQL - это система управления реляционными базами данных. В реляционной базе данные хранятся в отдельных таблицах, благодаря чему достигается выигрыш в скорости и гибкости. Таблицы связываются между собой при помощи отношений, благодаря чему обеспечивается возможность объединять при выполнении запроса данные из нескольких таблиц.

MySQL поддерживает SQL (структурированный язык запросов) и может применяться в качестве SQL-сервера. Это означает, что общаться с сервером можно на языке SQL: клиент посылает серверу запрос, тот его обрабатывает и отдает клиенту только те данные, которые были получены в результате этого запроса.

Кроме того, MySQL - это ПО с открытым кодом, т.е. его можно свободно изучать и изменять. Пакет распространяется на условиях GPL (General Public License), его можно бесплатно загрузить из Интернета для некоммерческого применения.

С появлением Интернет-технологий, позволяющих создавать динамичные Web-страницы, необычайно возрос спрос и на СУБД, которые наиболее полно подходили бы для этого по быстродействию, надежности и стабильности. И здесь хорошо проявил себя пакет MySQL, который получился быстрым, простым и надежным.

По большому счету, отсутствие некоторых функций, которые были принесены в жертву быстродействию и надежности, не создает больших хлопот пользователям. Для работы с полноценной корпоративной базой данных MySQL неотягивает, но с повседневными задачами MySQL справляется довольно хорошо.

Каждый из этих критериев может быть охарактеризован через набор определенных параметров. Рассмотрим каждую из выделенных групп критериев в отдельности. С чего начать оптимизацию. После определения, какие запросы выполняются при генерации страницы. Возможны такие варианты:

1. Есть тяжелые запросы, занимающие сотни миллисекунд.
2. Запросов много, но все они выполняются достаточно быстро.

В первом случае необходимо попробовать оптимизировать отдельные запросы. Здесь поможет SQL-оператор EXPLAIN и знания об индексах. Это решение применимо ко всем сайтам, в том числе размещенным на виртуальном хостинге.

Во втором случае имеет смысл заняться углубленным анализом логов и тонкой настройкой MySQL. На виртуальном хостинге сделать это не получится, только на VPS и выделенных серверах.

Приложение 6

Пример выполнения заданий ЛР 7

```
/* Авторы без повторений */  
SELECT DISTINCT Book.Author  
FROM Book;
```

```
/* Авторы без повторений с сортировкой */  
SELECT DISTINCT Book.Author FROM Book  
ORDER BY Book.Author;
```

```
/* Без повторения авторы и годы, сортировка по автору по возрастанию, а среди записей с одинаковым  
автором сортировка по году издания книг по убыванию */  
SELECT DISTINCT Book.Author, Book.BookYear  
FROM Book  
ORDER BY Book.Author, Book.BookYear DESC;
```

```
/* Книги от самых старых к самым новым. Книги с одинаковым годом издания упорядочиваются по  
названию в алфавитном порядке. */  
SELECT Book.*  
FROM Book  
ORDER BY Book.BookYear, Book.Title;
```

```
/* список из двух самых новых изданий */  
SELECT TOP 2 *  
FROM Book  
ORDER BY Book.BookYear DESC;
```

```
/* информация книгах, составляющих первую половину общего списка из таблицы Book,  
упорядоченного по возрастанию года */
```

```
SELECT TOP 50 PERCENT *  
FROM Book  
ORDER BY Book.BookYear;
```

```
/* выборка библиотечных номеров книг, их авторов и названий */  
SELECT BookInLib.LibID, Book.Author, Book.Title  
FROM Book INNER JOIN BookInLib ON Book.BookID = BookInLib.BookID;
```

```
/* выборка библиотечных номеров книг, их авторов и названий LEFT JOIN*/  
SELECT BookInLib.LibID, Book.Author, Book.Title  
FROM Book LEFT JOIN BookInLib ON Book.BookID = BookInLib.BookID;
```

```
/* запрос на выборку книг, выпущенных с 1997 по 2001 год */  
SELECT Book.Author, Book.Title, Book.Publisher, Book.BookYear  
FROM Book  
WHERE Book.BookYear BETWEEN 1997 and 2001;
```

```
/* запрос на выборку книг, выпущенных с 1997 по 2001 год и принадлежность издательства заранее  
определённому списку*/  
SELECT Book.Author, Book.Title, Book.Publisher, Book.BookYear  
FROM Book  
WHERE Book.BookYear BETWEEN 1997 AND 2001  
AND Book.Publisher IN ('Политехника', 'Азбука');
```

```
/* эквивалентно предыдущему */  
SELECT Book.Author, Book.Title, Book.Publisher, Book.BookYear  
FROM Book  
WHERE (((Book.Publisher)='Азбука' Or (Book.Publisher)='Политехника') AND ((Book.BookYear) Between  
1997 And 2001));
```

/*запрос, который выведет в обратном алфавитном порядке названия книг, выпущенных после 1999 года, фамилия автора (значение поля Author) у которых начинается с буквы «Г» или название издательства заканчивается буквой «а». LIKE в синтаксисе Access!!! */

```
SELECT Book.Author  
FROM Book  
WHERE BookYear > 1999 and (Author like 'Г*' OR Publisher like '*а')  
ORDER BY Book.Author DESC;
```

/* неиспользуемые статусы книг */

```
SELECT BookStatus.StatusName  
FROM BookStatus LEFT JOIN BookInLib ON BookStatus.StatusID = BookInLib.StatusID  
WHERE BookInLib.StatusID Is Null;
```

/* Последовательность логических операторов. LIKE в синтаксисе Access!!! */

```
SELECT *  
FROM Book  
WHERE Author LIKE 'А*' OR BookYear> 2000 AND NOT Publisher LIKE '[И,П]*';
```

Приложение 7

Построение многомерной модели хранилища данных киоска данных для анализа продаж компании «Mebele_Life»

Для решения поставленных задач необходима интегрированная информационная база данных. После анализа предметной области, была построена модель с помощью инструмента Erwin. Это обосновано тем, что данное case-средство позволяет добавлять компоненты, такие как сущности, связи (идентифицирующие и не идентифицирующие), дает возможность создания идентификаторов, простых атрибутов и репозитория (т.е. метаданных), что в свою очередь позволяет описывать UDP, Data Movement Sours, Data Movement Rules и т.д. Также Erwin предлагает выбор СУБД, которые поддерживают многомерную модель и возможность построения модели на физическом уровне, что позволяет использовать нотацию Dimensional, специально предназначенную для разработки хранилищ данных. Модель хранилища данных киоска Продажи в схеме типа снежинка представлена на рисунке 7.1.

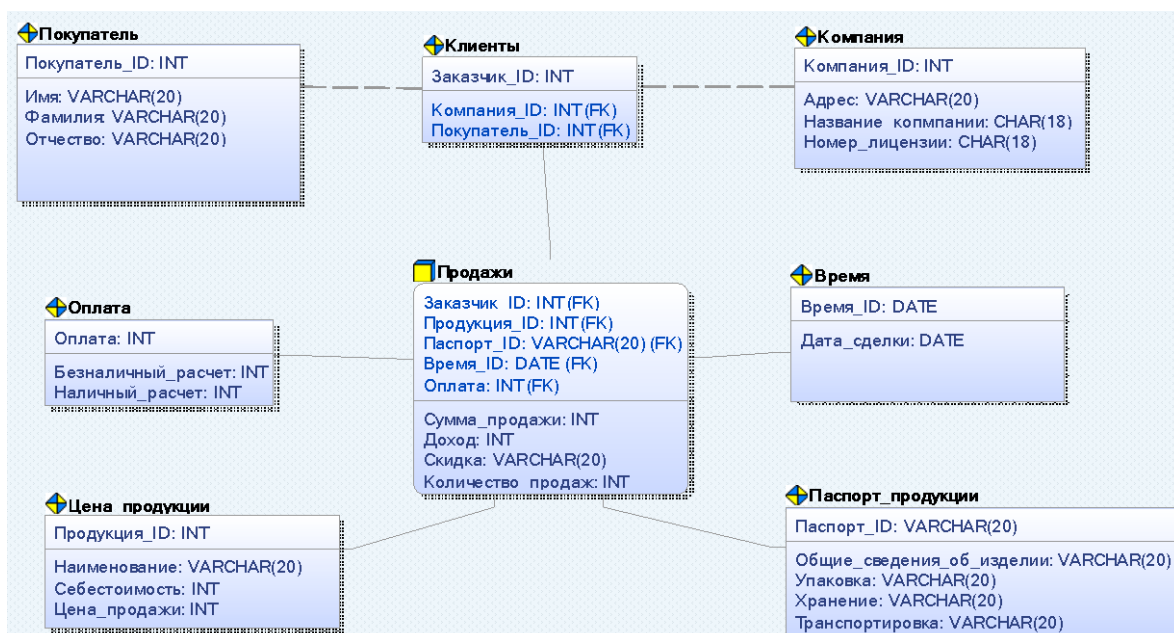


Рис. 7.1. Модель хранилища данных в схеме типа снежинка для анализа продаж компании «Mebele_Life»

Модель построена в виде звезды, которая является стандартом в размерном моделировании. Схема звезда содержит в себе одну большую таблицу – факт и таблицы размерности (деменсиал), которые соединяются с таблицей факта в виде звезды.

Учебное текстовое электронное издание

Махмутова Марина Владимировна

УПРАВЛЕНИЕ ДАННЫМИ

Учебно-методическое пособие

1,40 Мб

1 электрон. опт. диск

г. Магнитогорск, 2021 год

ФГБОУ ВО «МГТУ им. Г.И. Носова»

Адрес: 455000, Россия, Челябинская область, г. Магнитогорск,
пр. Ленина 38

ФГБОУ ВО «Магнитогорский государственный
технический университет им. Г.И. Носова»

Кафедра бизнес-информатики и информационных технологий

Центр электронных образовательных ресурсов и
дистанционных образовательных технологий

e-mail: ceor_dot@mail.ru