

Министерство образования Республики Беларусь
Учреждение образования «Белорусский государственный университет
информатики и радиоэлектроники»

Факультет компьютерных систем и сетей
Кафедра информатики
Дисциплина «Архитектура вычислительных систем»

«К ЗАЩИТЕ ДОПУСТИТЬ»

Руководитель курсового проекта,
ассистент

_____. А. В. Давыдчик
_____. _____. 2023

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к курсовому проекту

на тему:

**«ВЕБ-ПРИЛОЖЕНИЕ ДЛЯ ОРГАНИЗАЦИИ РАБОТЫ И
ВНУТРЕННИХ ДАННЫХ АВТОВОКЗАЛА»**

БГУИР КП 1-40 04 01 001 ПЗ

Выполнил студент группы 053504
БЕЛЬКО Владислав Игоревич

(подпись студента)

Курсовой проект представлен на
проверку _____. _____. 2023

(подпись студента)

Минск 2023

ВВЕДЕНИЕ

В современном мире эффективная организация работы и эффективное управление данными являются ключевыми факторами успеха для различных организаций. Автовокзалы, как важное звено в системе общественного транспорта, также нуждаются в современных технологических решениях, которые обеспечат эффективную работу и управление внутренними данными.

Целью данной курсовой работы является разработка веб-приложения, которое позволит автовокзалам оптимизировать свою работу и обеспечить более эффективное управление внутренними данными, такими как расписание автобусов, информация о поездках, билетная система и т. д.

В процессе разработки будут использованы современные подходы и методы моделирования данных, а также системы управления данными, которые обеспечат надежность, безопасность и гибкость функционала предлагаемого веб-приложения.

Эта курсовая работа представляет собой описание процесса разработки, выбранных методологий, исследования рынка и анализа требований, а также реализацию прототипа веб-приложения. Ожидается, что результаты данной работы будут полезны для автовокзалов, стремящихся повысить эффективность своей работы и обеспечить более качественное обслуживание для своих клиентов.

1 ТЕОРЕТИЧЕСКИЕ ОСНОВЫ МОДЕЛЕЙ ДАННЫХ И СИСТЕМ УПРАВЛЕНИЯ БАЗАМИ ДАННЫХ

1.1 Модели данных

[Сюда текст о понятии модели данных]

1.1.1 Иерархическая модель данных

Иерархические базы данных — это самая ранняя модель представления сложной структуры данных. Информация в иерархической базе организована по принципу древовидной структуры, в виде отношений «предок-потомок». Каждая запись может иметь не более одной родительской записи и несколько подчиненных. Связи записей реализуются в виде физических указателей с одной записи на другую. [1]

Графически такую структуру можно изобразить в виде дерева, состоящего из объектов различных уровней (рисунок 1.1). Верхний уровень занимает один объект, второй — объекты второго уровня и так далее.

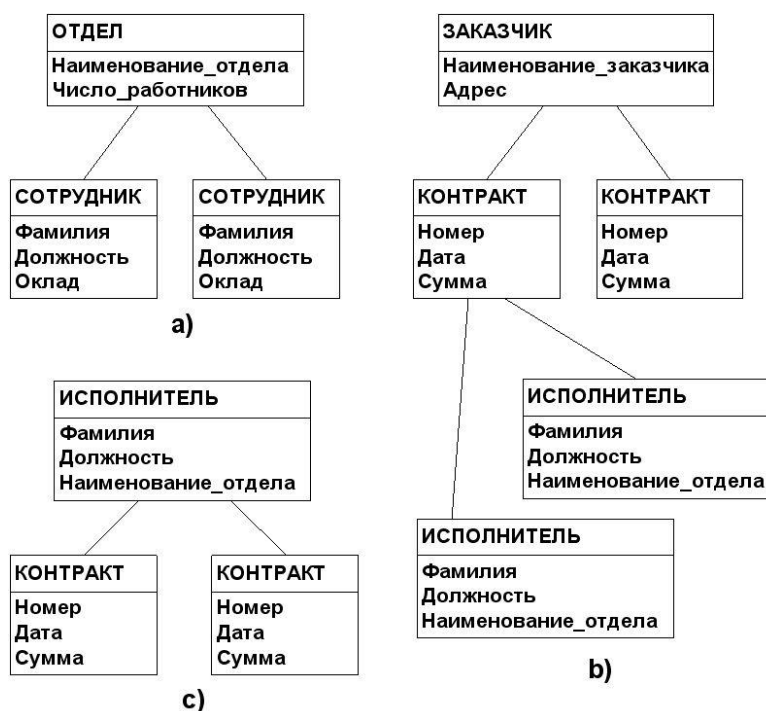


Рисунок 1– пример иерархической базы данных

Между объектами существуют связи, каждый объект может включать в себя несколько объектов более низкого уровня. Такие объекты находятся в отношении предка (объект более близкий к корню) к потомку (объект более низкого уровня), при этом возможно, чтобы объект-предок не имел потомков или имел их несколько, тогда как у объекта-потомка обязательно только один предок. Объекты, имеющие общего предка, называются близнецами.

Примеры

К примерам иерархичных баз данных можно отнести:

- IBM DBOMP;
- Information Management System (IMS);
- Time-Shared Date Management System (TDMS);
- Mark IV MultiAccess Retrieval System;
- System 2000.

Недостатки

К основным недостаткам иерархических моделей следует отнести: неэффективность, невозможность реализации связей «многие-ко-многим», медленный доступ к сегментам данных нижних уровней иерархии, четкая ориентация на определенные типы запросов и др.

Также недостатком иерархической модели является ее громоздкость для обработки информации с достаточно сложными логическими связями, а также сложность понимания для обычного пользователя.

Иерархические СУБД быстро прошли пик популярности, которая обуславливалась их ранним появлением на рынке. Затем их недостатки сделали их неконкурентоспособными, и в настоящее время иерархическая модель представляет исключительно исторический интерес.

1.1.2 Сетевая модель данных

Сетевая модель данных является одной из классических моделей данных, которая была разработана в 1960-х годах. Эта модель представляет данные в виде нескольких взаимосвязанных и взаимодействующих между собой записей, образуя сеть, где каждая запись может иметь несколько родителей и несколько потомков (рисунок 1.2).

В сетевой модели данные представлены в виде записей или сегментов, которые хранят фактическую информацию. Каждая запись имеет свой уникальный идентификатор и может содержать набор полей, описывающих характеристики данных.

Разница между иерархической моделью данных и сетевой состоит в том, что в иерархических структурах запись-потомок должна иметь в точности одного предка, а в сетевой структуре данных у потомка может иметься любое число предков.

Сетевая БД состоит из набора экземпляров определенного типа записи и набора экземпляров определенного типа связей между этими записями.

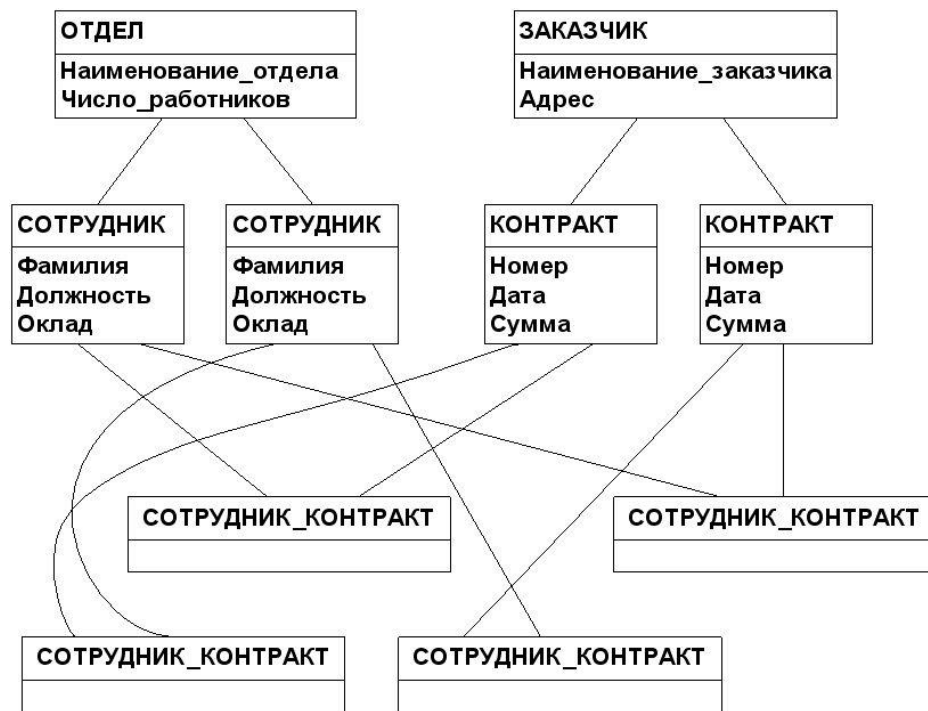


Рисунок 1.2 – Пример сетевой базы данных

Примеры

К примерам иерархичных баз данных можно отнести:

- Integrated Data Store (IDS);
- DMS-1100;
- CDC DMS-170;
- Burroughs Data Management System (DMS-2)
- IMAGE/3000

Достоинства

- Гибкость. Сетевая модель позволяет представлять сложные отношения между записями и обеспечивает более гибкую структуру данных, чем иерархическая модель;
- Эффективность. Сетевая модель обеспечивает быстрый доступ к данным, так как связи между записями оптимизированы для быстрого поиска и обработки;
- Целостность данных. Сетевая модель позволяет задавать ограничения целостности данных, такие как ограничения на связи между записями.

Недостатки

- Сложность. Сетевая модель может быть сложной для понимания и поддержки вследствие сложности связей и сетевой структуры данных;

- Зависимость от физической структуры. Структура данных в сетевой модели связана с физической структурой базы данных, что может создавать проблемы при изменении или модификации данных;
- Ограниченная интеграция. Интеграция с другими системами и моделями данных может быть сложной в сетевой модели, особенно в сравнении с реляционной моделью данных;

1.1.3 Реляционная модель данных

Реляционная модель данных – это набор данных с predetermined связями между ними.

Эти данные организованы в виде набора таблиц, состоящих из столбцов и строк. В таблицах хранится информация об объектах, представленных в базе данных. В каждом столбце таблицы хранится определенный тип данных, в каждой ячейке – значение атрибута.

Каждая строка таблицы представляет собой набор связанных значений, относящихся к одному объекту или сущности.

Каждая строка в таблице может быть помечена уникальным идентификатором, называемым первичным ключом, а строки из нескольких таблиц могут быть связаны с помощью внешних ключей.

К этим данным можно получить доступ многими способами, и при этом реорганизовывать таблицы БД не требуется.



Рисунок 1.3 – Пример реляционной базы данных

Примеры

- Oracle;
- MySQL;
- Microsoft SQL Server;
- PostgreSQL.

Достоинства

К достоинствам реляционных СУБД можно отнести:

- Наличие развитой теории реляционной модели данных. Реляционная модель данных поддерживается в большей степени, чем другие модели. Для манипулирования отношениями используются элементы реляционной алгебры;
- Возможность и наличие аппарата сведения других моделей данных к реляционной;
- Поддержка специальных средств ускоренного доступа к данным. Имеются прежде всего индексирование и кэширование;
- Возможность манипуляции данными без необходимости знаний физической организации базы данных;
- Простота использования реляционной модели данных.

Благодаря своим достоинствам реляционная модель данных получила широкое распространение в таких традиционных бизнес приложениях, как обработка заказов, учет складских запасов, банковское дело, заказ авиабилетов, бухгалтерский учет и т.д.

Недостатки

Среди других недостатков реляционной модели отмечают:

- Слабое представление сущностей реального мира;
- Семантическая перегрузка;
- Слабая поддержка ограничения целостности;
- Однородная структура данных;
- Ограниченный набор операций;
- Трудности организации рекурсивных запросов.

СПИСОК ЛИТЕРАТУРЫ

[1]

https://www.tadviser.ru/index.php/%D0%A1%D1%82%D0%B0%D1%82%D1%8C%D1%8F:%D0%98%D0%B5%D1%80%D0%B0%D1%80%D1%85%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B0%D1%8F_%D0%A1%D0%A3%D0%91%D0%94