

# Понятие информационной системы, БД и СУБД,

## Лекция №1

Бутенко И.В. 2016 год

# Структура курса

## MS SQL Server 2014

- 2 вводные лекции
- 10 лекций + лаб. Работы
  
- Разработка информационных систем 3  
лекции + лаб. работы
- Теоретические основы реляционной модели 3  
лекции + лаб. работы
- Экзамен

# MS SQL

- ИС, БД, СУБД
- Модели.
- Производители СУБД. MS SQL.
- Таблицы
- SELECT, INSERT, UPDATE, DELETE
- Транзакции и блокировки
- ХП, Функции, Курсоры, Представления, Триггеры
- Производительность, Администрирование

# Требования

## Зачет:

- Лабораторные работы (1, 2, 3)
- Задание по реляционной алгебре

## Экзамен:

- Допуск – наличие зачета

# Экзамен

- Время подготовки к экзамену 30 минут
- В экзаменационном билете содержится 3 вопроса
- Преподаватель имеет право дать несколько дополнительных вопросов
- При подготовке к экзамену можно пользоваться любым материалом

# Введение

Естественная человеческая потребность в потреблении информации повлекла за собой ее сохранение на различных носителях.

Основные изобретения:

- письменность
- перфокарты
- магнитные ленты
- жесткие магнитные диски

# Области применения ВТ

Области использования вычислительной техники:

- применение ВТ, в качестве средств коммуникации;
- применение ВТ для выполнения численных расчетов, выполнение которых вручную слишком долго или вообще невозможно;
- использование средств ВТ в автоматических или автоматизированных ИС.

# Основные определения

- **ИС** –ориентированная на конечного пользователя система для сбора, хранения и обработки информации. Система называется информационной, если она поддерживает информационную поддержку бизнеса.
- **База данных** - это набор структурированной информации, предназначенный для совместного использования несколькими пользователями одновременно.

# Сравнение ИС и СРВ

| <i>Тип программной системы</i>    | <i>Информационные системы</i> | <i>Системы реального времени</i> |
|-----------------------------------|-------------------------------|----------------------------------|
| Структуры данных                  | Сложные                       | Простые                          |
| Объем входных данных              | Большой                       | Малый                            |
| Функциональный акцент             | Интенсивный ввод/вывод        | Интенсивные вычисления           |
| Требования к аппаратной платформе | Машинная независимость        | Машинная зависимость             |
| Требования на время отклика       | Мягкое реальное время         | Жесткое реальное время           |

# Файловые системы

- **Файл** – совокупность связанной информации.
- **Файл** – именованная область внешней памяти, в которую можно записывать и из которой можно считывать данные.

Можно выполнять стандартные файловые операции:

- создать файл;
- открыть ранее созданный файл;
- прочитать из файла некоторую запись (текущую, предыдущую, следующую, первую, последнюю);
- записать в файл на место текущей новую или добавить в конец;
- удалить запись;
- удалить файл.

# Пример

| Сотрудники    |
|---------------|
| Имя           |
| Должность     |
| Отдел         |
| Дата рождения |
| Зар. плата    |

| Проекты    |
|------------|
| Название   |
| Стоимость  |
| Сотрудники |

# Функции БД

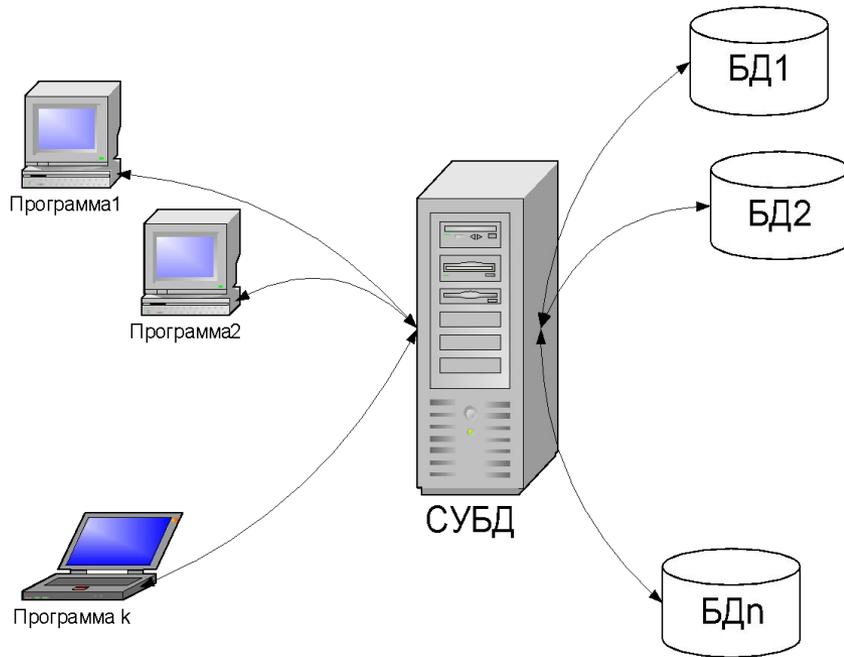
- Данные должны быть *структурированными*, мы должны знать о *связях* между экземплярами данных и поддерживать *целостность* данных.
- Должна поддерживаться *многопользовательская* работа с данными.
- Должна поддерживаться *транзакционность*.
- Должны быть *методы разграничения* доступа к данным.

**Транзакция** – это набор действий воспринимаемый базой данных как единое целое. Действия в транзакции либо выполняются все либо откатываются все.

# Системы управления БД

- **СУБД или DBMS (database management system)** - это компьютеризированная система хранения информации, целью которой является хранить информацию и предоставлять ее по требованию.

# Назначение СУБД



- СУБД – промежуточный уровень между прикладной программой (ИС) и данными.
- Назначением СУБД является упрощение разработки и как следствие возможность более просто создавать качественные ИС.

# Функции СУБД

- Управление данными во внешней памяти
- Управление буферизацией оперативной памяти
- Управление транзакциями
- Журнализация
- Поддержка языков БД

# Пользователи СУБД

- Программисты
- Администраторы БД
- “Продвинутые” пользователи БД
- Прикладные программы

# Классификация СУБД

По многопользовательской работе имеются:

- однопользовательские СУБД (single-user system);
- многопользовательские СУБД (multi-user system).

По оперативности обработки информации:

- СУБД, поддерживающие OLTP (On Line Transaction Processing);
- СУБД, поддерживающие аналитическую деятельность и многомерное представление данных OLAP (On Line Analysis Processing).

# Архитектура СУБД

- Ядро СУБД (Database Engine)
- Компилятор языка БД (обычно SQL)
- Набор утилит

# Типы утилит

- Утилиты управления БД:
  - создание баз данных
  - создание объектов баз данных
  - резервное копирование и восстановление
  - перенос данных.
- Средство для написания запросов на языке БД.
- Утилиты для проверки производительности.
- Набор средств для конфигурирования связи между базами данных и прикладными программами.

# Архитектура ИС

- Файл-сервер
- Хост-терминал
- Клиент-сервер
- Трехуровневая архитектура

# Файл-серверная архитектура (на основе разделяемых файлов)



- Особая популярность в конце 80-х годов
- На каждом из персональных компьютеров запускается приложение, использующее общие файлы, находящиеся на файловом сервере.

# Файл-серверная архитектура

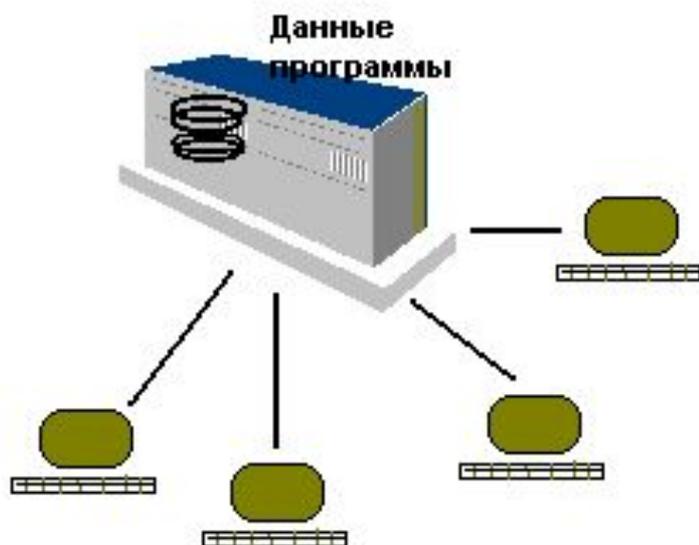
## Преимущества:

- Быстро и недорого можно запустить однопользовательское приложение в многопользовательском режиме.

## Недостатки:

- Нагрузка на трафик;
- Целостность данных;
- Если открыли файл на коррекцию, то другие могут только читать (квазимногопользовательская работа);
- При аппаратном сбое сервер не может проверить корректность данных, т.е. ничего не знает о структуре;
- При аппаратном сбое нарушение транзакционности.

# Архитектура «хост-терминал»



На центральном компьютере работает общее, единое для всех пользователей приложение.

Это приложение работает со своими данными.

Каждый из пользователей подключается к ИС через систему удаленного терминального доступа.

На рабочем месте пользователя производится прием нажатых клавиш, их пересылка на компьютер, получение и отработка команд на вывод информации.

# Архитектура «хост-терминал»

## Преимущества

- Многопользовательский доступ;
- Нет проблем с пропускной способностью, т.к. передавалась только та информация, которую мог воспринять и ввести пользователь.

## Недостатки

- Очень дорогое администрирование и сопровождение;
- Системы обеспечивали только алфавитно-цифровой монохромный интерфейс;
- Масштабируемость.

# Архитектура «клиент-сервер»



- Программа-клиент (Кл) - “активная” программа. В ее задачи входит генерация некоторых обращений за услугами к Сrv.
- Программа-сервер (Сrv) - пассивная программой. В ее функции входит ожидание запроса от Кл.
- Когда запрос поступает, Сrv обрабатывает его и, при необходимости, возвращает Кл некоторые результаты.

# Функции сервера

- Выполнять клиентские запросы по извлечению и модификации данных;
- Обеспечивать одновременный доступ к данным нескольких пользователей;
- Обеспечивать идентификацию пользователей и разграничение прав доступа разных пользователей к разным данным;
- Обеспечивать целостность и непротиворечивость данных в случае аппаратных и программных сбоев;
- Защищать данные от несакционированного доступа;
- Предоставлять дополнительные средства администрирования информационной системы.

# Функции клиента

- Взаимодействие с внешним миром
  - Экранные формы
  - Диалоги
  - Системные задачи

# Архитектура «клиент-сервер»

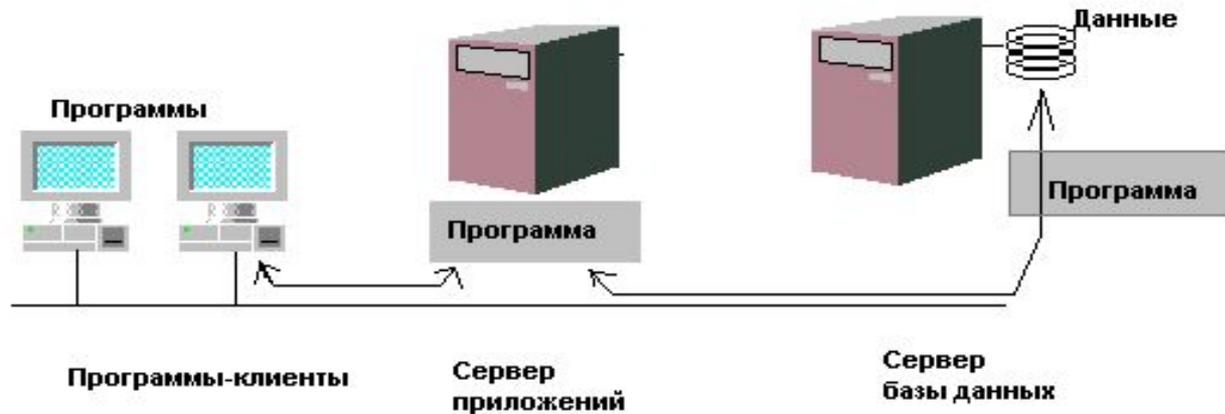
## **Толстый клиент**

- Как правило, сервер является лишь хранилищем данных, а вся работа по обработке и представлению этих данных переносится на машину клиента.

## **Тонкий клиент**

- Возможность работы в терминальном режиме.

# Трехуровневая архитектура



Разбиваем приложение на две части:

- собственно клиент
- сервер данного приложения.

# Трехуровневая архитектура

## **Преимущества**

- Масштабируемость
- Конфигурируемость
- Высокая безопасность
- Высокая надежность

## **Недостатки**

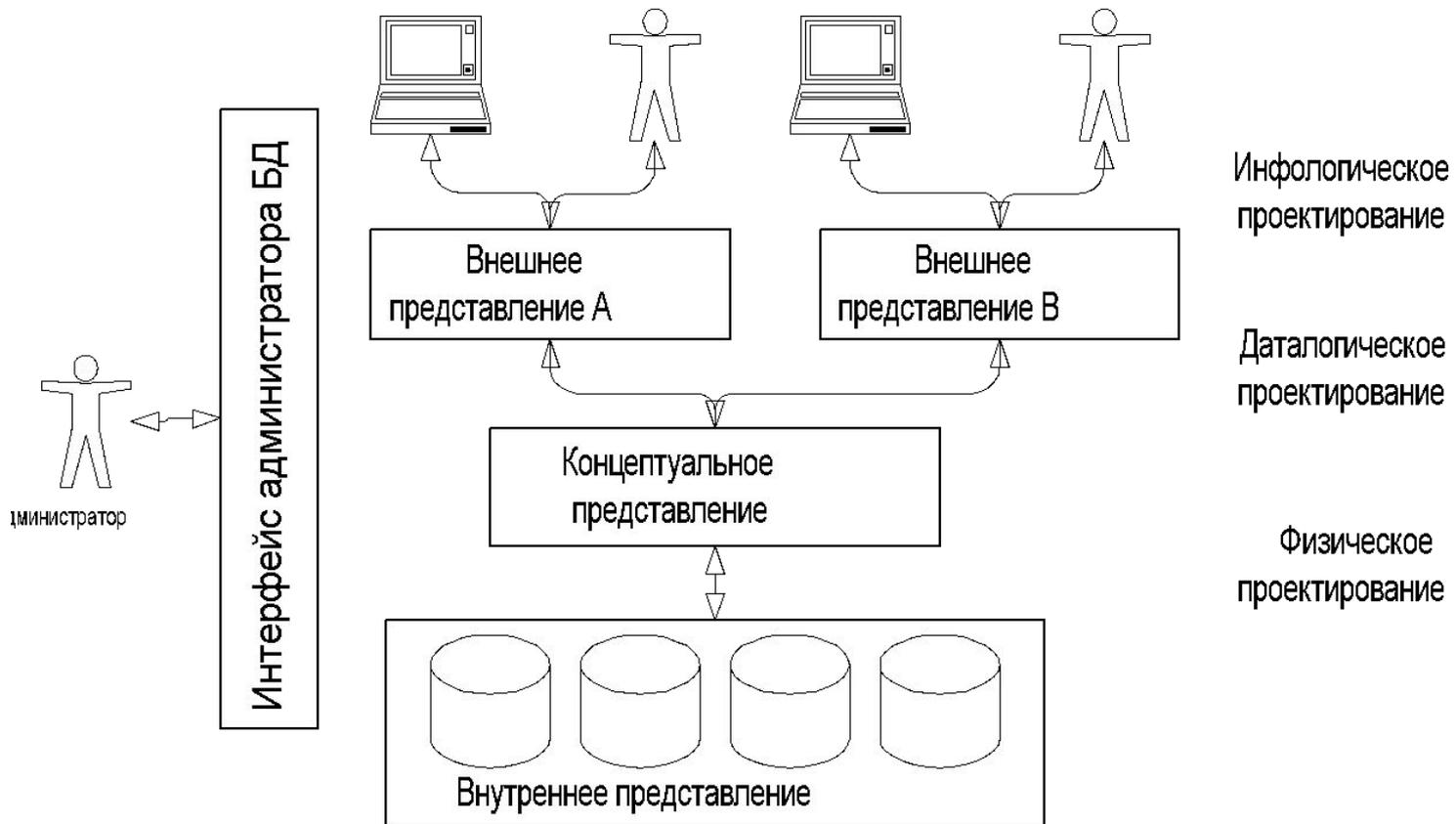
- Высокая сложность создания приложений
- Сложность в разворачивании и администрировании

# Слои логики

Архитектура приложения определяется способом, которым программные компоненты распределены в приложении, и, таким образом, является одной из самых главных характеристик системы, напрямую влияющих на другие ее свойства и параметры.

- Компоненты уровня представлений
- Прикладные компоненты (бизнес-компоненты)
- Компоненты доступа к информационным ресурсам

# Архитектура ИС



# Архитектура ИС

- Инфологическая модель предметной области - описание предметной области, выполненное без ориентации на используемые в дальнейшем СУБД и технические средства.
- Даталогическая модель базы данных - отображение логических связей между информационными элементами ИМ.
- Физическая модель - определяет используемые запоминающие устройства, способы физической организации данных в среде хранения.

# Уровни восприятия данных

- Уровень конечного пользователя - прикладной (пользовательский);
- Уровень программиста и администратора – концептуальный;
- Уровень реализации - физический уровень.

Реализация физического уровня – производители СУБД: MSSQL, Oracle DB, IBM DB2, Sybase (комм.), MySQL, PostgreSQL.

# ЯЗЫКИ СУБД

- SDL – Schema Definition Language
- DDL – Data Definition Language
- DML – Data Manipulation Language

# Резюме

- Потребность в сохранении и повышении ценности информации → БД.
- Проблемы файловых систем → СУБД.
- СУБД – связующее звено между БД и прикладными программами.
- Клиент-серверная архитектура: Кл - “активная” программа, Срв – пассивная.