Основные понятия БД и СУБД.

Учебные вопросы.

1. **История развития и архитектуры систем управления базами данных.**
2. **Основные понятия и определения теории баз данных.**

1. История развития СУБД

- В 1968 году была введена в эксплуатацию первая промышленная СУБД – система IMS фирмы IBM
- <u>Первый этап</u>

В 1975 году появился первый стандарт СУБД, разработанный ассоциацией по языкам систем обработки данных – Conference of Data System Language (CODASYL).

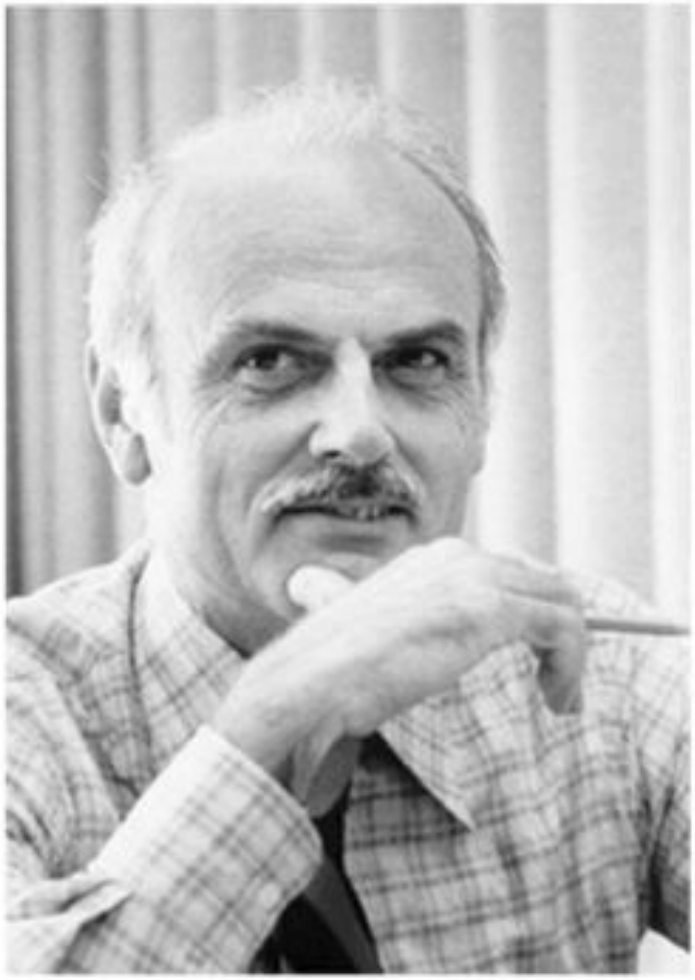
*Особенности 1-го этапа развития СУБД (***Файлы и файловые системы***):*

- все СУБД базировались на мощных мультипрограммных ОС (*RSX, UNIX*);
- функции управления распределением ресурсов в основном осуществлялись ОС;
- поддерживались языки низкого уровня манипулирования данными;
- значительная роль отводилась администрированию данных;

В 1981 году Э.Ф. Кодд создал реляционную модель данных и применил к ней операции реляционной алгебры. В истории развития баз данных можно выделить следующие этапы:

- Файлы и файловые системы;
- Базы данных на больших ЭВМ. Первые СУБД;
- Эпоха персональных компьютеров. Настольные СУБД;
- Распределенные базы данных.

Эдгар Франк Кодд
(Edgar Frank Codd)



Дата рождения: **23 августа** **1923**

Место рождения: **Портланд** (Дорсет)

Дата смерти: **18 апреля** **2003** (79 лет)

Страна: **Англия**

Научная сфера: **информатика**

Альма-матер: **Оксфордский университет**

Известен как: Создатель **реляционной модели** данных.

MyShared

- Второй этап (80-е года XX века) развития СУБД связан с организацией БД на больших машинах типа IBM 360/370, ЕС-ЭВМ, мини-ЭВМ типа PDP11 (фирмы *Digital Equipment Corporation — DEC*) и на разных моделях HP (фирмы *Hewlett Packard*).

- создана первая система (*System R*), реализующая идеологию реляционной модели данных;
- введено понятие **транзакции** (группа логически объединённых последовательных операций по работе с данными, обрабатываемая или отменяемая целиком);
- появлялись первые языки высокого уровня для работы с реляционной моделью данных, однако стандарты для этих языков отсутствовали.

Третий этап (90-е года XX века) связан с внедрением персональных компьютеров - *ПЕРСОНАЛИЗАЦИЯ*.

- все СУБД были рассчитаны на создание БД в основном с монопольным доступом, в редких случаях предполагалась последовательная работа нескольких пользователей;
- большинство СУБД имели развитый и удобный пользовательский интерфейс и инструментарий для разработки готовых приложений без программирования;

- при наличии высокоуровневых языков манипулирования данными типа реляционной алгебры и SQL (*Structured Query Language*) в настольных СУБД поддерживались низкоуровневые языки манипулирования данными на уровне отдельных строк таблиц;

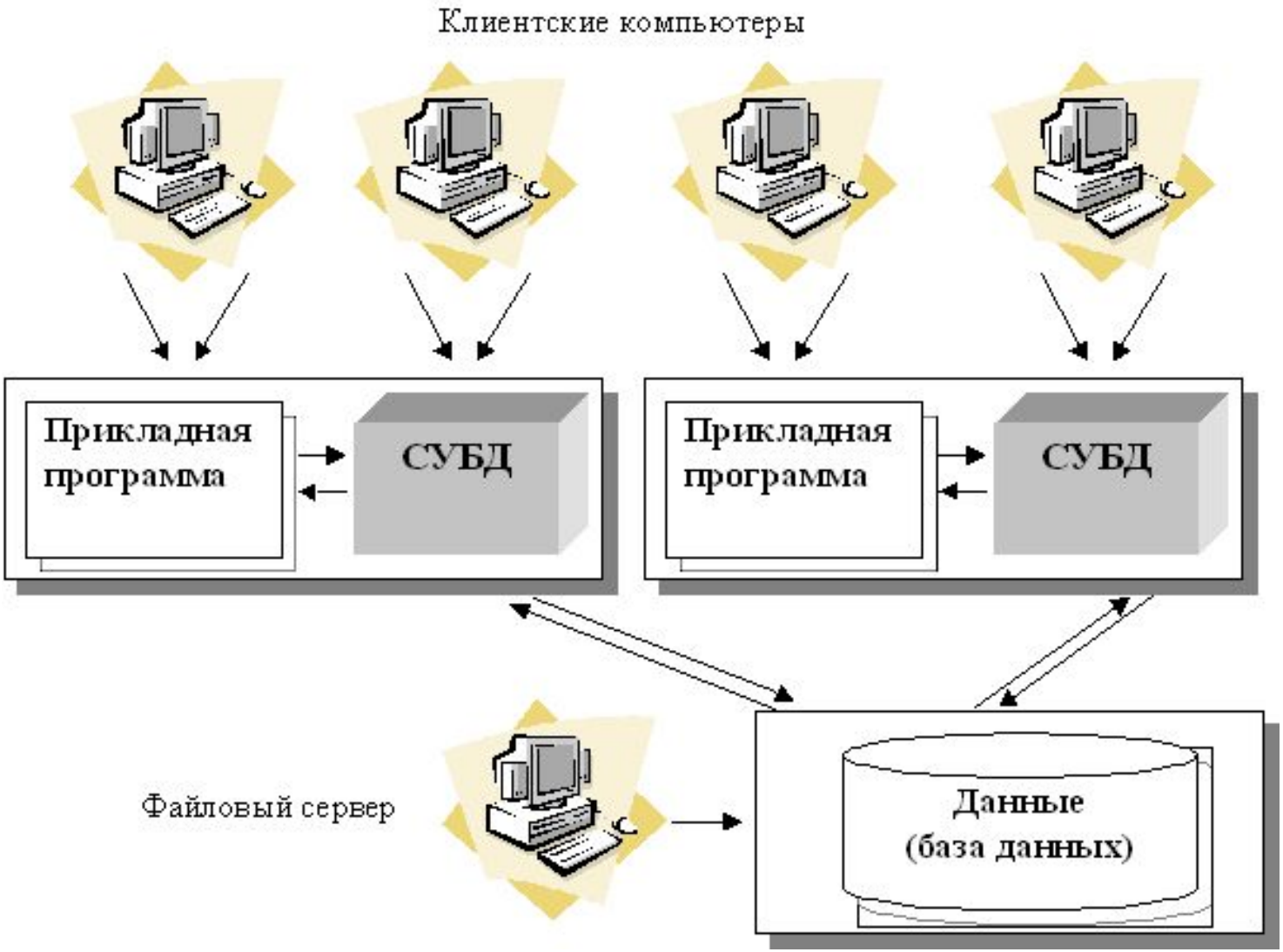
Примеры настольных СУБД - *dBase*, *FoxPro, Clipper, Paradox*.

Решение этих задач привело к появлению распределенных данных, сохраняющих все преимущества настольных СУБД и в то же время позволяющих организовать параллельную обработку информации и поддержку целостности БД.

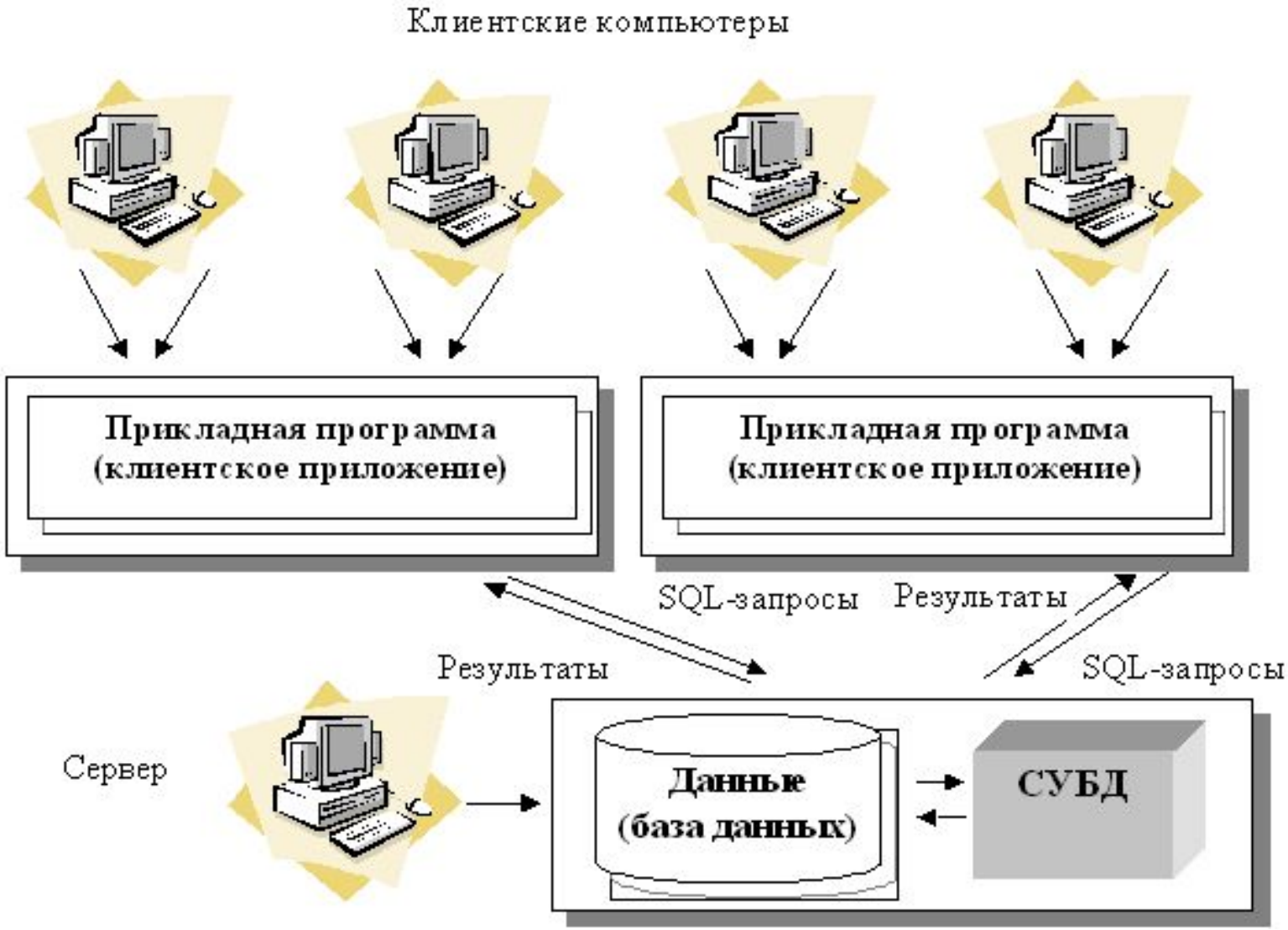
1.1. ***Архитектурные решения доступа к данным в компьютерных сетях***

1. Централизованная архитектура

2. Архитектура "файл-сервер"

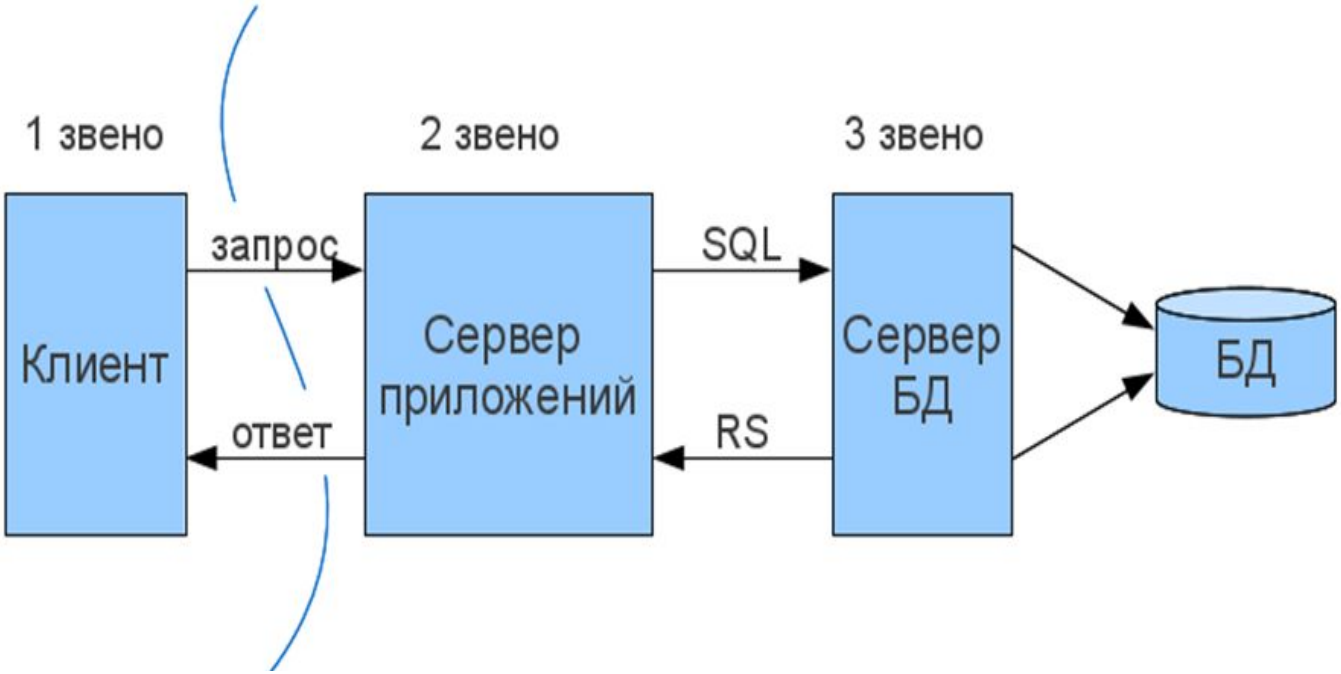


3. Архитектура "клиент-сервер“.

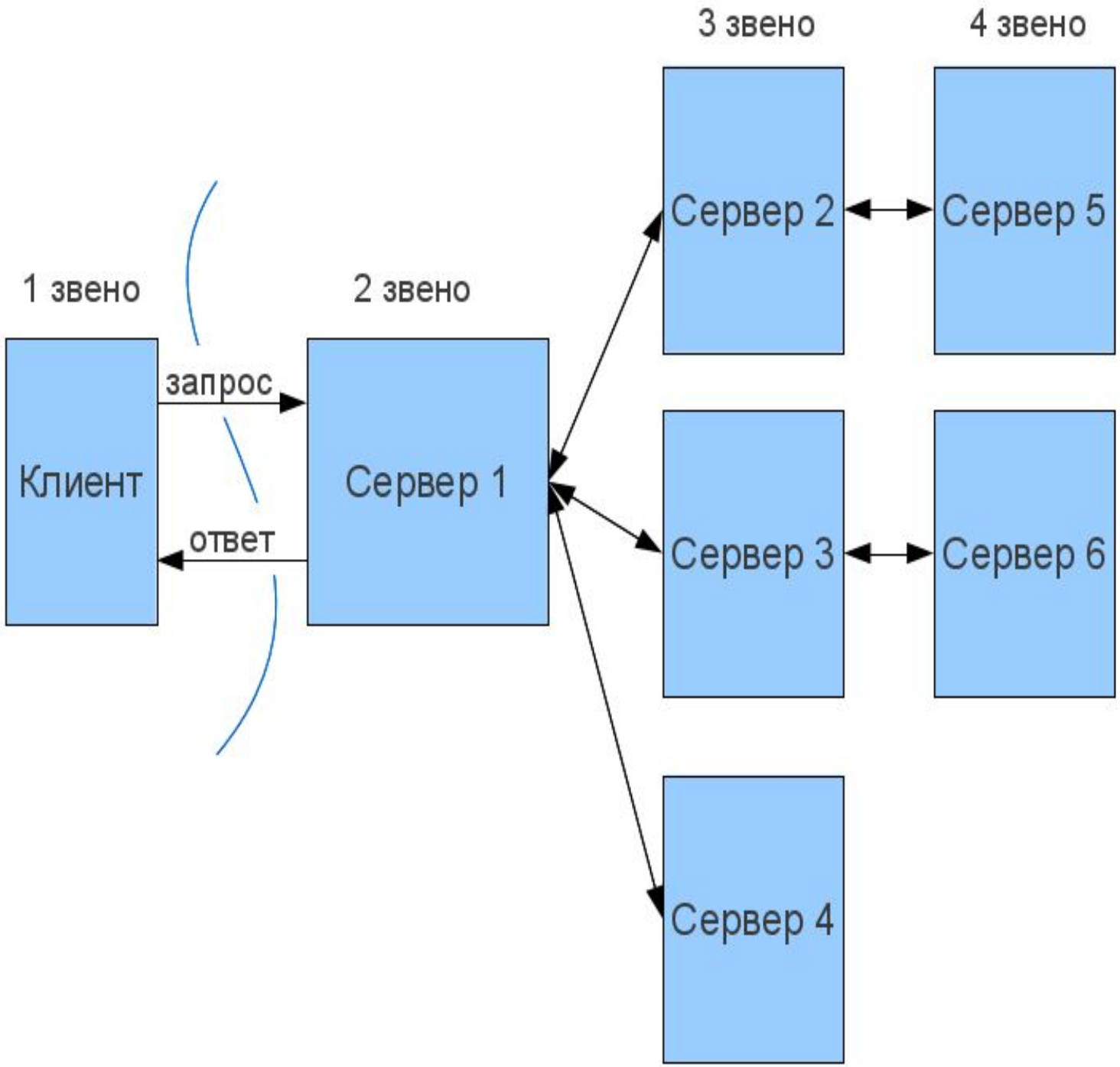


Архитектура «клиент-сервер» определяет общие принципы организации взаимодействия в сети, где имеются *серверы*, узлы-поставщики некоторых специфичных функций (сервисов) и *клиенты*, потребители этих функций.

Практические реализации **архитектуры «клиент-сервер»** называются **клиент-серверными технологиями**.



Представление данных — на стороне клиента.
Прикладной компонент — на выделенном сервере приложений (как вариант, выполняющем функции промежуточного ПО).
Управление ресурсами — на сервере БД, который и представляет запрашиваемые данные.


3 звено
4 звено
Сервер 2
Сервер 5
1 звено
2 звено
запрос
Клиент
Сервер 1
ответ
Сервер 3
Сервер 6
Сервер 4

Клиент-серверные технологии

(сервисы и серверы):

- Web-серверы
- Серверы приложений
- Серверы баз данных
- Файл-серверы
- Прокси-сервер
- Файрволы (брандмауэры)
- Почтовые серверы
- Серверы удаленного доступа (RAS) *(Remote Access Service)*

Особенности 4-го этапа развития СУБД - ИНТЕГРАЦИЯ:

- большинство современных СУБД рассчитаны на многоплатформенную архитектуру, т.е. они могут работать на компьютерах с разной архитектурой и под разными ОС;
- разработаны ряд стандартов в рамках языков описания данных и языков манипулирования данными, например *SQL,* и технология по обмену данными между различными СУБД, например протокол *ODBC* (*Open DataBase Connectivity*) фирмы Microsoft.

- Представителями СУБД, относящимися к этому этапу, являются *MS Access* и все современные серверы баз данных - *Oracle, MS SQL, Informix, DB2, SQL Base* и др.

Четвертый этап характеризуется использованием сети Интернет.

- Основное отличие этого подхода от технологии клиент-сервер состоит в том, что в этом случае нет необходимости использования специализированного клиентского ПО.
- Для работы с удаленной БД используется стандартный браузер Интернета.

- При этом встроенный в загружаемые пользователем HTML-страницы код, написанный обычно на языках *Java, Java-script, Perl* и др., отслеживает все действия пользователя и транслирует их в низкоуровневые SQL-запросы к БД, выполняя таким образом ту работу, которой в технологии клиент-сервер занимается клиентская программа.

- Однако беспрецедентные объемы данных заставляют разработчиков и бизнес использовать альтернативы реляционных баз данных, работающих более тридцати лет. В совокупности все эти технологии известны как «*NoSQL базы данных*». Термин «NoSQL» был придуман Эриком Эвансом (Eric Evan / Racker).

Классификация СУБД.

- *По языкам общения* СУБД делятся на *открытые* (используются универсальные языки программирования), *замкнутые* (собственные языки общения с пользователями) и *смешанные*.
- *По числу уровней в архитектуре* различают *одноуровневые*, *двухуровневые*, *трехуровневые* системы.
- *По сфере возможного применения* различают *универсальные* и *специализированные*, обычно проблемно-ориентированные СУБД.

- СУБД поддерживают разные типы данных.
- *Набор типов данных*, допустимых в разных СУБД, различен. В настоящее время наблюдается тенденция к расширению числа используемых типов данных. Кроме того, ряд СУБД позволяет разработчику (прикладному программисту или администратору БД) добавлять новые типы данных и новые операции над этими данными. Такие системы называются *расширяемыми системами баз данных* (РСБД).

Функции системы управления базами данных.

1. Определение структуры создаваемой базы данных, ее инициализация и проведение начальной загрузки.
2. Предоставление пользователям возможности манипулирования данными (выборка необходимых данных, выполнение вычислений, разработка интерфейса ввода/вывода, визуализация).
3. Обеспечение независимости прикладных программ и данных (логической и физической независимости).
4. Защита логической целостности базы данных.
5. Защита физической целостности.
6. Управление полномочиями пользователей на доступ к базе данных.
7. Синхронизация работы нескольких пользователей.
8. Управление ресурсами среды хранения.
9. Поддержка деятельности системного персонала.

Транзакция – это законченный блок обращений к базе данных, для которого гарантируется выполнение четырех условий ACID:

- Атомарность (**A**tomicity);
- Согласованность (**C**onsistency);
- Изолированность (**I**solation);
- Долговременность (**D**urability).

2. Основные понятия и определения теории баз данных

- **Предметная область** - отражение в БД совокупности и объектов реального мира с их связями, относящимися к некоторой области знаний и имеющих практическую ценность для пользователя.
- **Объектом** называется элемент ИС, сведения о котором хранятся в БД. Объект называют <u>сущностью</u> (от англ, **entity**) - например, *таблица*.

- **Атрибут** - это информационное отображение свойств объекта. Каждый объект характеризуется некоторым набором атрибутов – *столбец, поле*.
- **Элемент данных** - содержимое одной ячейки атрибута – *содержание ячейки в столбце*.
- **Запись данных** (англ. record) - это совокупность значений связанных элементов данных - *строка*.
- **Ключевым элементом данных** называются такой атрибут (или группа атрибутов), который позволяет определить значения других элементов-данных – *столбец/столбцы в таблице, позволяющий однозначно определить запись*.

- **Первичный ключ** - это атрибут (или группа атрибутов), который уникальным образом идентифицируют каждый экземпляр объекта (запись).
- **Вторичным ключом** называется атрибут (или группа атрибутов), значение которого может повторяться для нескольких записей (экземпляров объекта). Прежде всего вторичные ключи используются в операциях поиска записей.

Вопросы:

1.