

Учебный курс
**Технологии и средства разработки
корпоративных систем**

Лекция 1

Открытые системы. Клиент и сервер

Лекции читает

кандидат технических наук, доцент

Зыков Сергей Викторович

Особенности архитектуры "клиент-сервер" для СУБД

Архитектура "клиент-сервер" для СУБД:

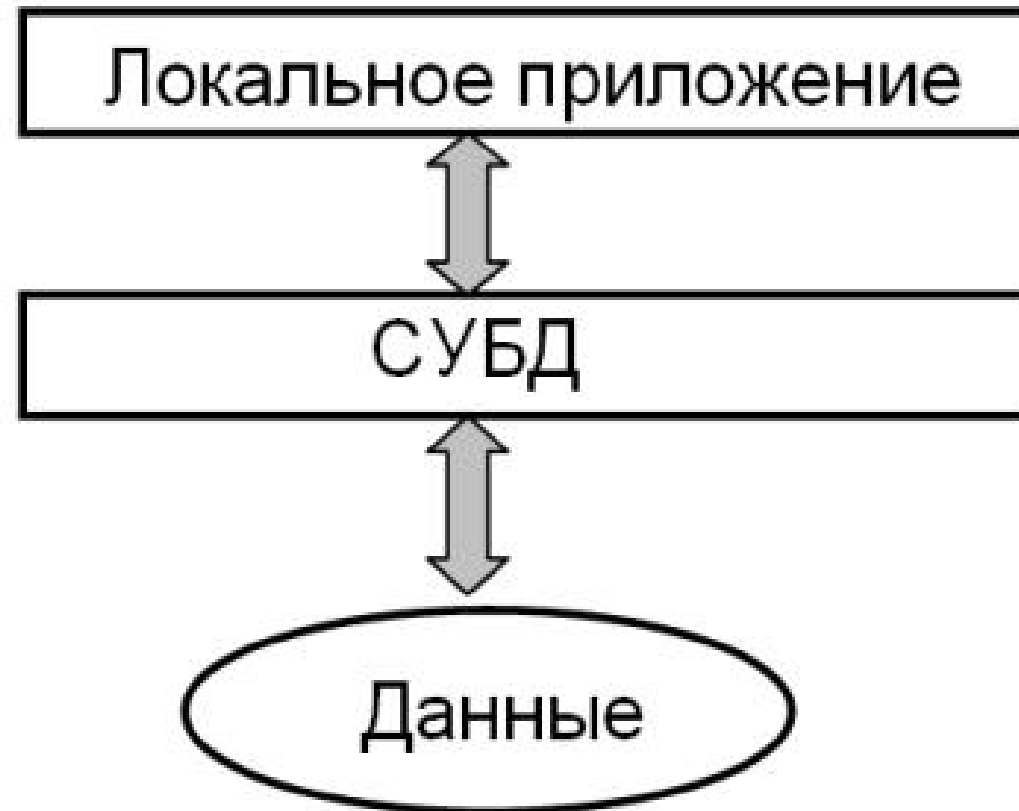
- разделение системы на приложение-клиент (front-end) и сервер БД (back-end)
- графический интерфейс пользователя
- Легкость и скорость разработку приложений за счет централизованного хранения ограничений целостности на сервере

Преимущества - синтез лучших черт :

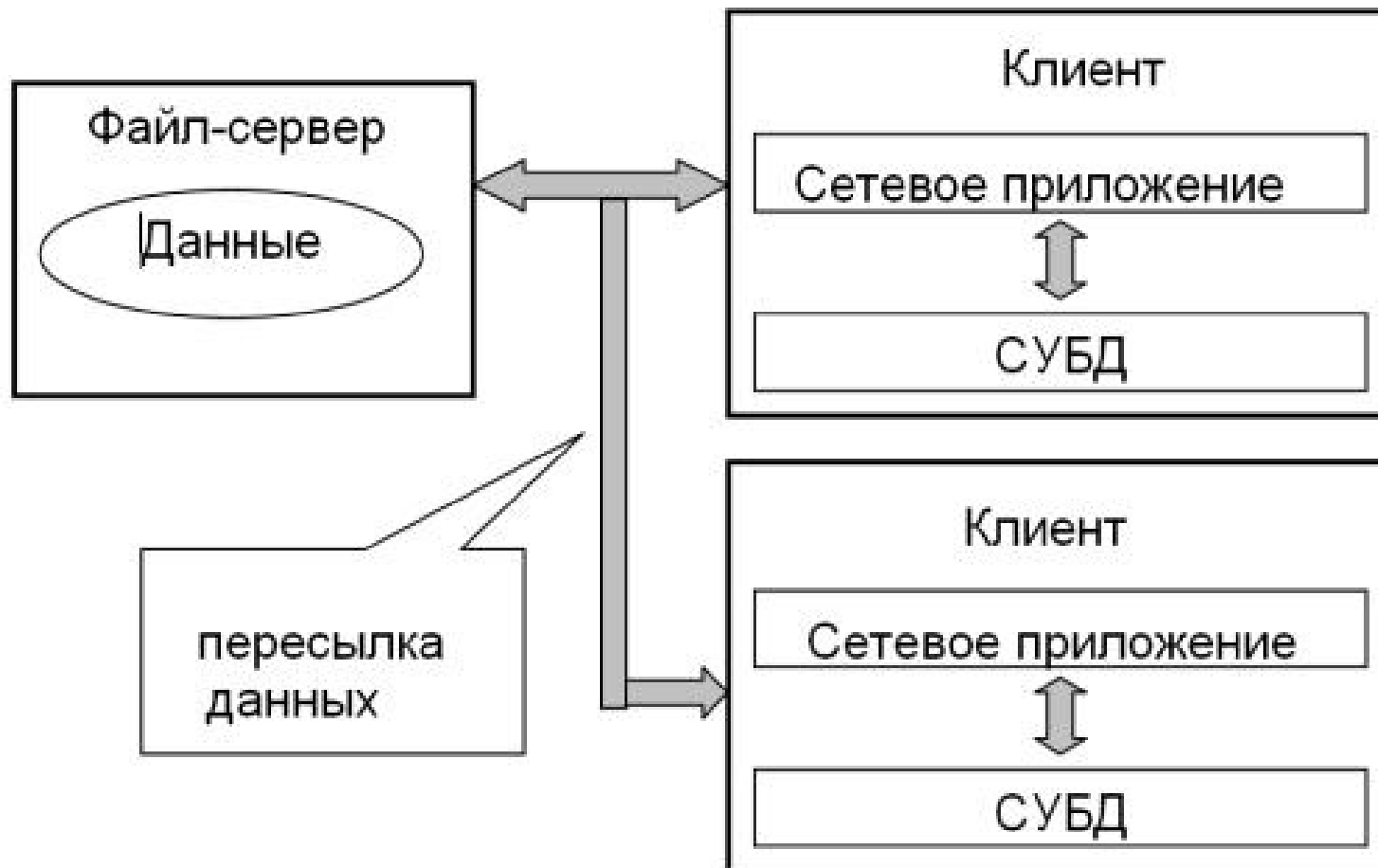
- мэйнфреймов:
 - централизованное администрирование,
 - безопасность,
 - надежность.
- "файл-серверов":
 - низкая стоимость
 - распределенная обработка данных на основе ресурсов «клиентов»

Корпоративные системы
Программные архитектуры распределенных СУБД

Обработка данных СУБД: локальный компьютер



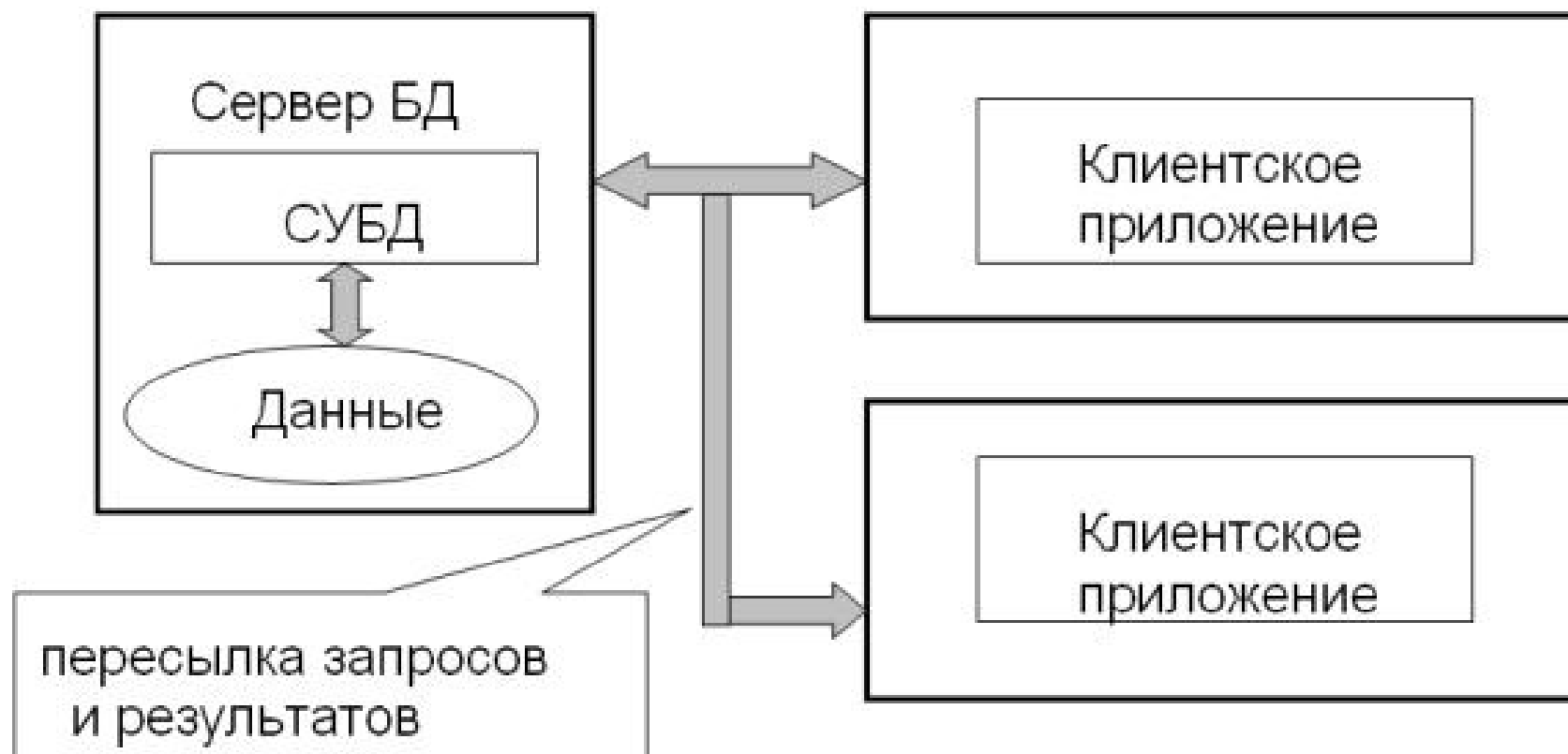
Корпоративные системы
Программные архитектуры распределенных СУБД
Обработка данных СУБД: файл-сервер



© 2000-2001 МГУИЭ

Корпоративные системы
Программные архитектуры распределенных СУБД

Обработка данных СУБД: клиент-сервер



Архитектура серверов БД

Сервер БД - СУБД на основе архитектуры «клиент-сервер»

Особенности:

- Содержит серверную и клиентскую части
- предназначены для хранения и обеспечения доступа к БД
- общая БД доступна всем пользователям ЛВС.
- облегчение и удешевление технологии распределенных БД

[БД часто хранятся в одном узле ЛВС и поддерживаются одним сервером]

Принципы клиент-серверного взаимодействия в СУБД

- Доступ к БД производится обращением к клиентской части.
- Основа интерфейса клиент-сервер - язык SQL (стандарт интерфейса СУБД в открытых системах)
- SQL-сервер - сервер БД на основе SQL.
- Преимущество SQL-серверов – стандартный интерфейс (в идеале любой SQL-клиент совместим с любым SQL-сервером)
- Недостаток SQL-серверов – сервер является узким местом системы (при этом клиент может быть достаточно мощным)
- Перспектива развития СУБД – гибкое распределение функций между клиентом и сервером

Преимущества RPC

- RPC важны в СУБД архитектуры "клиент-сервер".
- Балансировка функций клиентской и серверной частей (компоненты системы может располагаться на стороне как сервера, так и клиента).
- Сокращение различий между взаимодействующими компьютерами в физически неоднородной ЛВС
- Снижение важности аппаратно совместимых серверов и рабочих станций.

Разделение функций между клиентами и серверами

На стороне клиента СУБД работает ПО с доступом к БД лишь через (SQL-сервер) БД.

Организация на стороне клиента работы с "локальным кэшем" БД – частью БД, которая интенсивно используется клиентской приложением – возможна лишь посредством создания «клиентской» копии сервера БД

Перенос части приложения на сторону сервера для реализации его вычислительной мощности возможен при использовании RPC (если базовое ПО сервера – ОС и др. – это позволяет)

Виды архитектур сервера БД

Сервер БД – важная часть системы “клиент-сервер” (определяет ее успех)

Главное требование к серверу БД – max производительность (min времени выполнения запросов при max числе пользователей).

Основные архитектуры сервера БД:

- **Многопроцессная** - одновременная работа нескольких экземпляров исполняемого файла
- «+»: масштабируемость,
- «-» - ресурсоемкость по памяти – нужна для каждого экземпляра приложения
- **Многопоточная** - один исполняемый файл с несколькими потоками
- «+» - меньшая ресурсоемкость АО, отсутствие сложного взаимодействия процессов

Корпоративные системы
Программные архитектуры распределенных СУБД
Особенности архитектур сервера БД

Многопроцессная :

- эффективный механизм взаимодействия процессов;
- Ориентация на ОС при разделении процессорного времени между экземплярами
- Пользователь при подключении к СУБД запускает экземпляр исполняемого файла сервера БД.
- Пример: Oracle Server

Многопоточная :

- Сервер разделяет время между потоками, ранжируя задачи.
- Примеры: MS SQL Server, Sybase SQL Server.

Модели трехуровневой архитектуры «клиент-сервер» (1)

Компоненты верхнего уровня «клиент-сервер»:

- презентационная логика (Presentation Layer, PL) - работа с данными пользователя
- бизнес-логика (Business Layer, BL) - проверка корректности данных, поддержка ссылочной целостности и т.д.
- логика доступа к ресурсам (Access Layer, AL) - хранение данных

Отсюда модели взаимодействия «клиент-сервер»:

- "Толстый" клиент (или RDA - Remote Data Access) – объединение в клиентском приложении PL и BL при реализации AL серверной частью
- "Тонкий" клиент – клиент реализует PL, а сервер объединяет BL и AL
- Сервер приложения (бизнес-логики или AS - Application Server) – BL выделяется как особое приложение

Модели трехуровневой архитектуры “клиент-сервер” (2)

“Толстый” клиент – особенности:

- Наиболее распространен во внедренных и активно используемых системах
- децентрализация управления бизнес-логикой
- При необходимости изменений в клиентском приложении нужно менять исходный код
- Серверная часть с AL - сервер БД

“Тонкий” клиент – особенности:

- Распространением в корпорациях, на основе Web-технологий и браузеров
- Скромные требования клиента к АО
- Мах загрузка сервера за счет бизнес-логики на хранимых процедурах – откомпилированных SQL-инструкциях, хранящихся на сервере
- Централизация и легкость коррекции управления данными предприятия
- Безопасность и целостность данных

Модели трехуровневой архитектуры “клиент-сервер” (3)

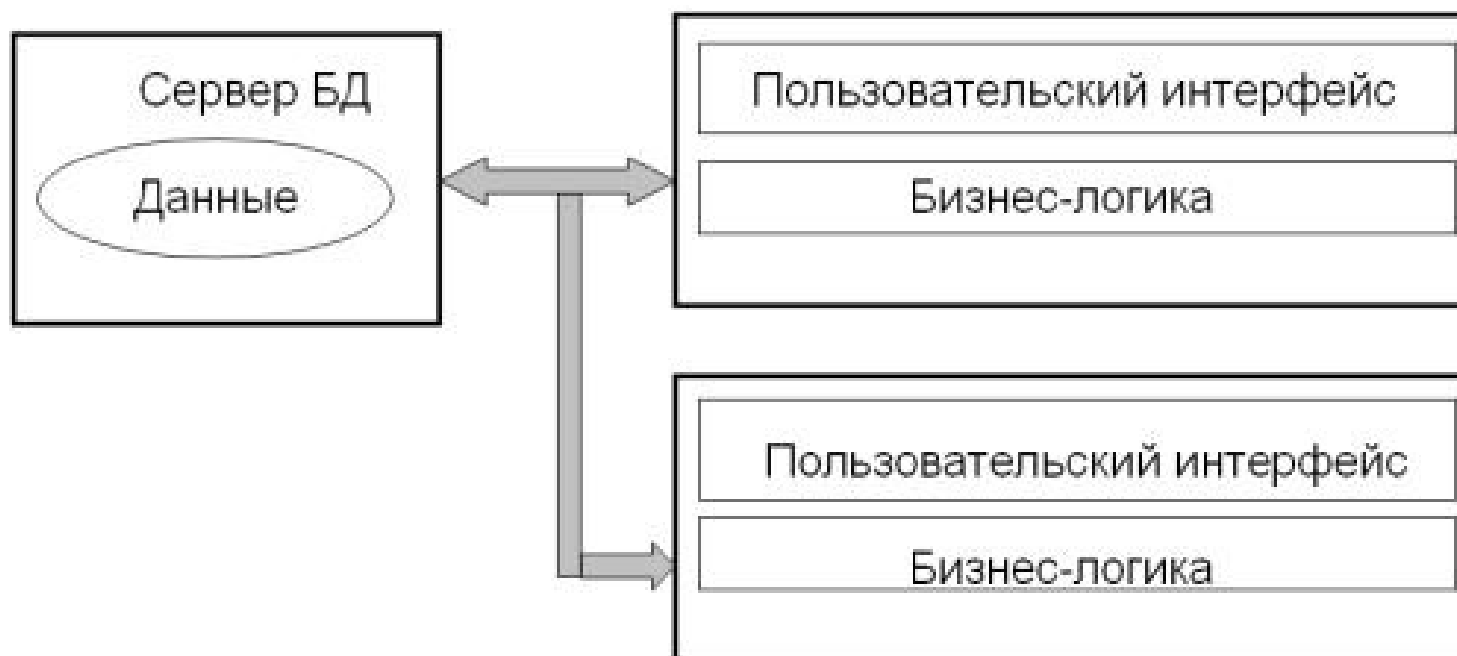
Сервер приложения – особенности:

- На сервере БД работает “универсальная” часть VL – бизнес-правила уровня предприятия или группы логически связанных приложений
- Поддержка «тонких» клиентов на компьютерах пользователей
- Освобождение сервера БД от чрезмерной загрузки
- Гибкие бизнес-правила
- Варианты сервера приложения:
 - SQL-сервер БД
 - Персональная СУБД (меньшие требования к АО; удобство поддержки VL)

Корпоративные системы
Программные архитектуры распределенных СУБД

Модели трехуровневой архитектуры «клиент-сервер»

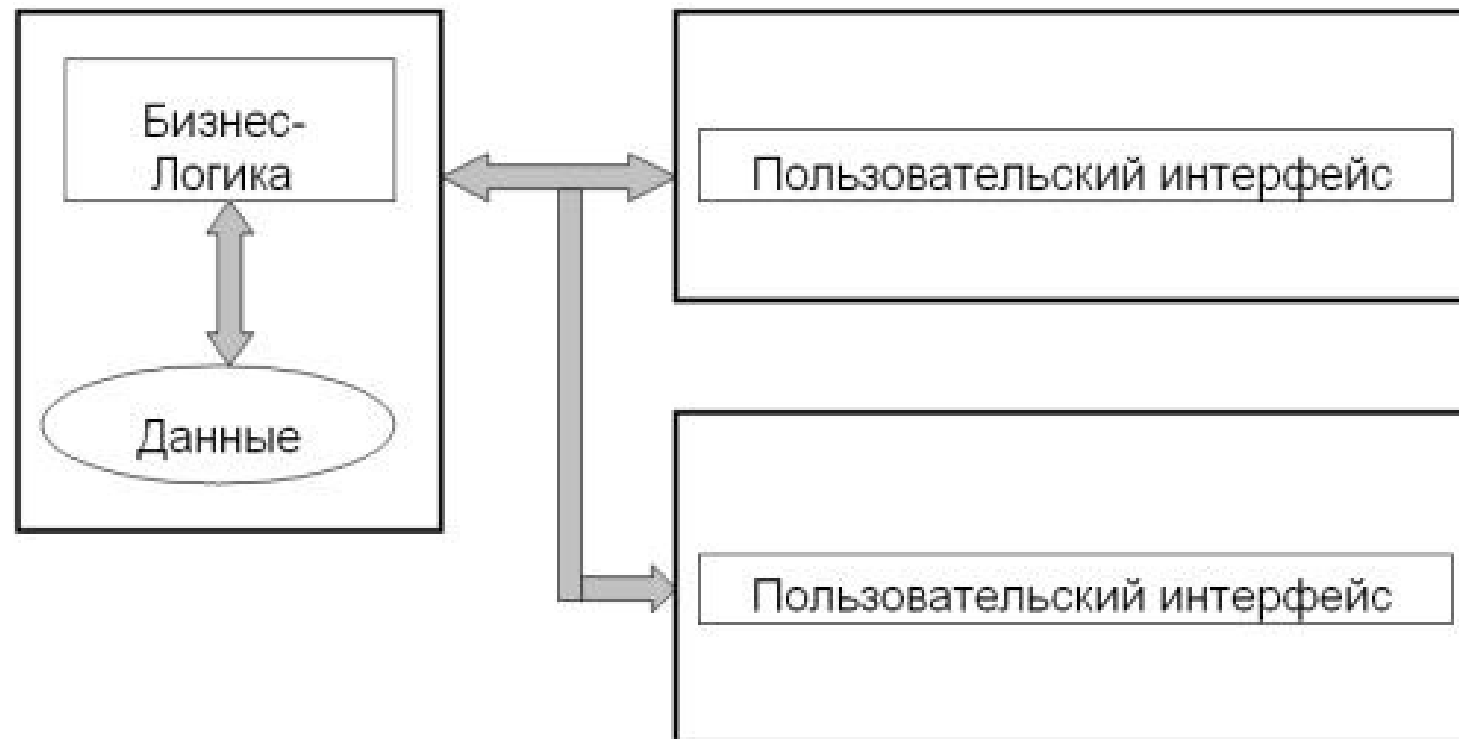
«Толстый» клиент:



Корпоративные системы
Программные архитектуры распределенных СУБД

Модели трехуровневой архитектуры «клиент-сервер»

«Тонкий» клиент:



Корпоративные системы
Программные архитектуры распределенных СУБД

Модели трехуровневой архитектуры “клиент-сервер”

Сервер приложения:

