

Учебный курс

**Технологии и средства разработки
корпоративных систем**

Лекция 2

**Виды и особенности клиент-серверных
систем с БД**

Лекции читает

кандидат технических наук, доцент

Зыков Сергей Викторович

Инструментальные средства поддержки разработки БД (1)

Универсальные средства разработки клиентских приложений

Функции:

- соединение с сервером БД
- разработка GUI для работы с данными.

Особенности:

- объектно-ориентированная среда (OLE, ODBC)
- языки 4GL
- Визуализация
- Командная разработка

Примеры: Oracle Developer, Borland Delphi, MS Visual Studio, Sybase PowerBuilder

Инструментальные средства поддержки разработки БД (2)

Персональные СУБД

Функции:

- Разработка бизнес-логики
- независимая работа клиентского приложения
- собственные форматы хранения данных

Особенности:

- Легкий и удобный GUI
- Интеграция с пакетами офисных приложений
- визуальные средства быстрой разработки
- средства быстрого построения приложений и отчетов
- объектно-ориентированная среда (OLE, ODBC)

Примеры: MS Access, Lotus Approach

Архитектура “клиент-сервер” в сетях (Internet / Intranet) (1)

1. Двухуровневая архитектура “клиент-сервер” - Особенности:

- Четкое разделение функций Web-браузера и Web-сервер
- Web-сервер предоставляет браузеру HTML-страницы
- Web-браузер отображает HTML-страницы, обрабатывая HTML-тэги

2. Трехуровневая архитектура “клиент-сервер” - Особенности:

- Клиент - браузер, сервер - сервер БД, промежуточный слой - Web-сервер (с программными расширениями)
- Протокол HTTP для передачи данных между браузером и сервером БД
- Преимущества:
 - Снижение сетевого трафика,
 - взаимозаменяемость компонент
 - Повышение безопасности.
- Недостаток:
 - Сложность обработки транзакций БД в силу не сохранения состояний HTTP

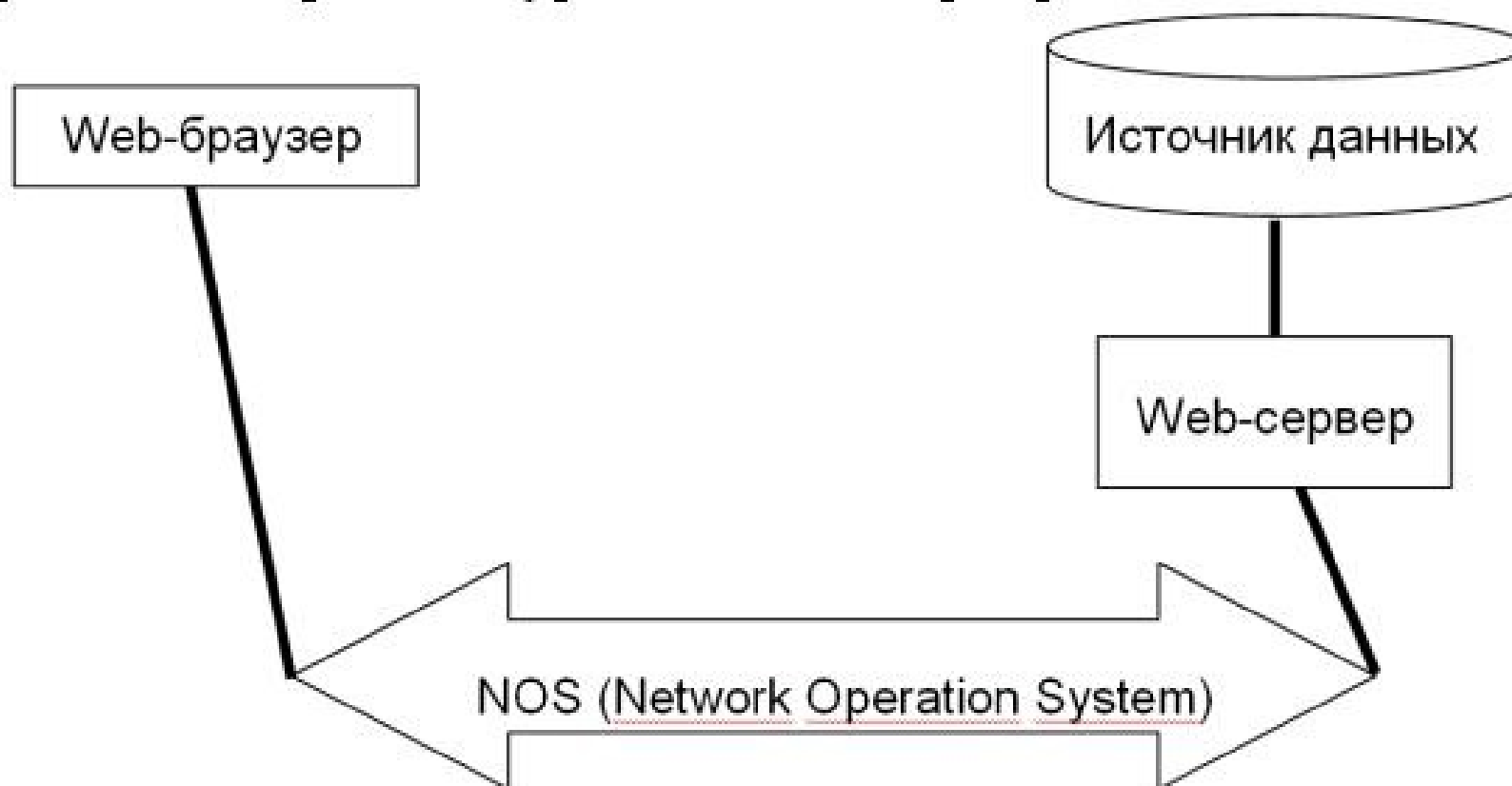
Архитектура “клиент-сервер” в сетях (Internet / Intranet) (2)

Трехуровневая архитектура “клиент-сервер” – порядок взаимодействия:

1. Браузер посылает Web-серверу запрос на доставку Web-страниц/данных
2. Web-сервер обслуживает заявки на Web-страницы, а запросы отправляет программе-расширению серверной части.
3. Программа-расширение принимает запрос, преобразует его и передает серверу БД.
4. Сервер БД обслуживает запрос и возвращает результат программе-расширению.
5. Программа-расширение преобразует результаты в формат браузера и передает их Web-серверу
6. Web-сервер передает результаты браузеру.

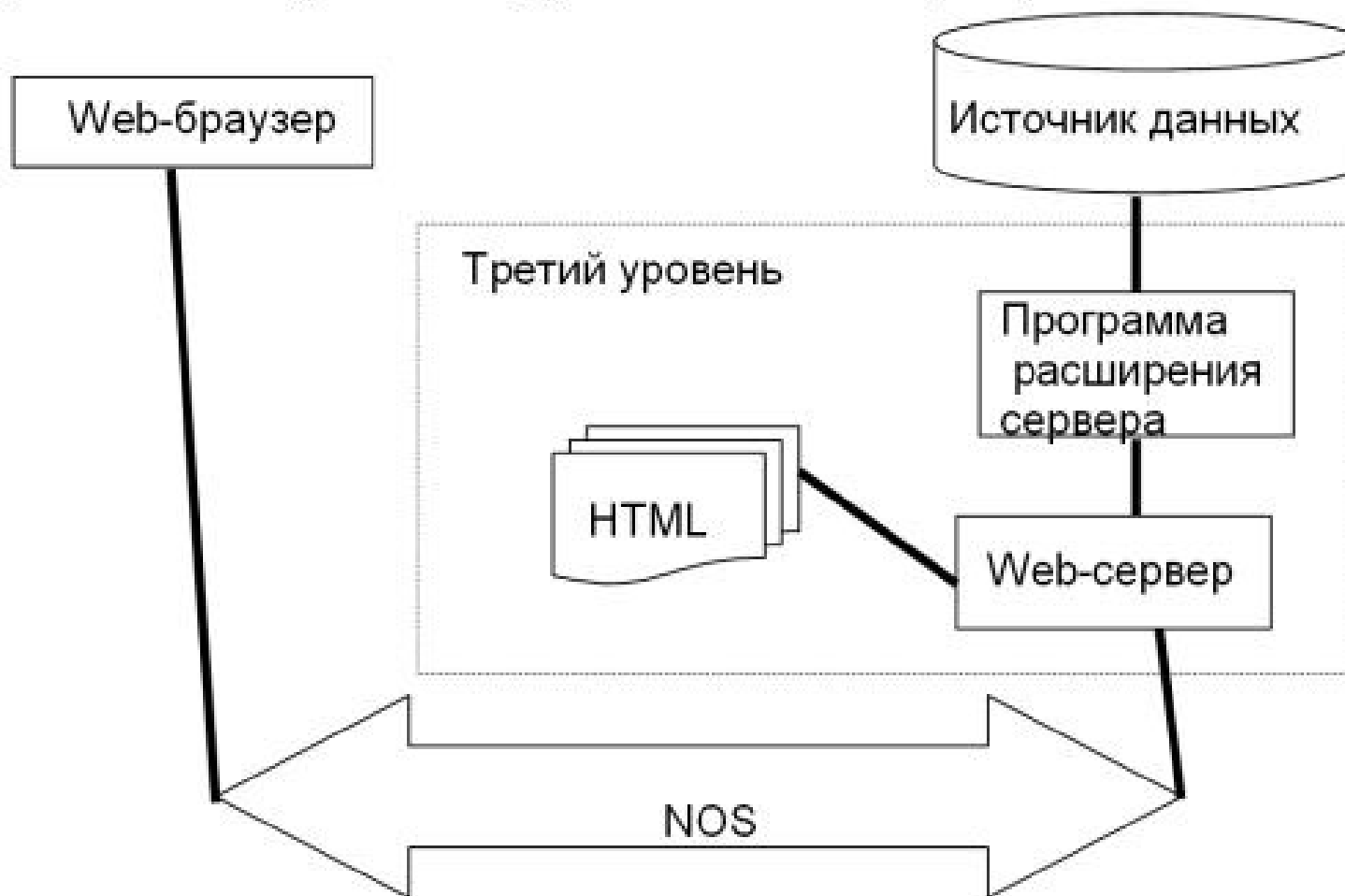
Архитектура “клиент-сервер” в сетях (Internet / Intranet) (3)

Двухуровневая архитектура “клиент-сервер”



Архитектура “клиент-сервер” в сетях (Internet / Intranet) (3)

Трехуровневая архитектура “клиент-сервер”



Программы расширения серверной части

Цель: преобразование данных между клиентом и сервером (на основе стандартов)

Задачи:

- Поддержка соединений БД для минимизации трафика в сети
- Поддержка резерва соединений БД для минимизации затрат на открытие/закрытие БД.
- Поддержка взаимозаменяемости/наращиваемости Web-серверов и серверов БД за счет стандартных интерфейсов

Типы расширений серверной части:

- Обычный CGI
- Гибридный CGI
- API

Интегрированные/федеративные системы и мульти-БД

Предпосылка: необходимость объединения систем БД на основе разных МД (расширений РМД, ООМД) и СУБД

Имеются теоретические рез-ты и практические реализации

Цель: интеграция неоднородных БД посредством:

- Разработки интегрированной МД для глобальной схемы БД;
- Преобразования операторов манипулирования данными глобальной БД в языки «локальных» СУБД
- Интеграции новых распределенных систем в глобальную среду (проблема – эффективность!):
 - Управление глобальными транзакциями
 - Сетевая оптимизация запросов
- Единообразного глобального администрирования данных

Мульти-БД – из-за требований пользователей сохраняют локальную автономность при частичной интеграции в федеративные системы (нет глобальной схемы данных, особые виды доступа к объектам и т.д.)

Grid-системы (1)

GRID - глобальная распределенная вычислительная сеть с производительностью, способной конкурировать с суперкомпьютерами

Предпосылки - потребность в решении вычислительных задач с огромными компьютерными мощностями и высокопроизводительными каналами передачи данных:

- биология (расшифровка генома человека)
- медицина (цифровая обработка и анализ изображения внутренних органов)
- геофизика (обработка информации, поступающей со спутников, следящих за состоянием Земли и атмосферы)
- физика элементарных частиц (моделирование и обработка экспериментов на существующих и строящихся ускорителях)

Ранее - суперкомпьютеры (дороговизна, плохая масштабируемость)

Особенности:

- поток информации ~100 Мбайт/с в течение ~10 лет
- большие вычислительные мощности для анализа данных (сотни тысяч современных ПК)

Корпоративные системы
Программные архитектуры распределенных СУБД
Grid-системы (2)

Особенности:

- Производительность, эффективность и качество работы сети за счет параллельных вычислений на большом количестве компьютеров, подключенных к сети
 - от тысяч до миллионов ПК,
 - оптоволоконные линии связи
- Десятки институтов, университетов и научных центров
 - Европейской Организацией по Ядерным Исследованиям = CERN, Швейцария
 - МИФИ - РФ

Перспективы - основа для развития технологий с высокопроизводительными каналами связи

- видеоконференции
- дистанционные формы обучения
- изучение фундаментальных процессов
- дистанционные научные исследования
- высокопроизводительные вычисления

Выводы

Основа распределенных СУБД – концепция открытых систем (стандартные интерфейсы, плавное наращивание функций/производительности)

Основа архитектур – идея разделения функций на клиентскую и серверную (с дальнейшей специализацией: телеком, БД, приложения и т.д.)

Компоненты архитектуры «клиент-сервер»:

- презентационная логика (PL),
- бизнес-логика (BL)
- логика доступа к ресурсам (AL)

Модели взаимодействия «клиент-сервер»:

- "Толстый" клиент (или RDA) = PL, BL на клиенте, AL на сервере
- "Тонкий" клиент – PL на клиенте, BL и AL - на сервере
- Сервер приложения (или AS) = BL - в особый слой

Интегрированные/федеративные/мульти-БД – «смесь» разных МД/СУБД

GRID - глобальная высокопроизводительная сеть