

Распределенные вычислительные системы

**Вводная лекция:
классификация и требования к
распределенным вычислительным
системам**

Алексей В. Бурдаков, к.т.н.
burdakov@usa.net

План лекций

№	Дата	Тема
1	28.10	<u>Вводная лекция: распределенные системы</u>
2	04.11	
3	11.11	
4	18.11	
5	25.11	
6	02.12	
7	09.12	
8	16.12	
9	23.12	

План лекции

- Архитектура вычислительной системы
- Распределенные системы
- Эволюция объектной технологии
- Литература

Архитектура вычислительной системы

- Представление о совокупности функциональных компонентов системы, их взаимосвязи и взаимодействии (интерфейсах). [Когаловский]
- Представление о модели, структуре и функционировании вычислительной системы, включающее характеристики индивидуальных компонентов и их взаимосвязи.
- Дизайн, структура, и поведение компьютерных систем, включая характеристики индивидуальных компонентов и способов их взаимодействия. [Encarta]

Классификация архитектур ВС

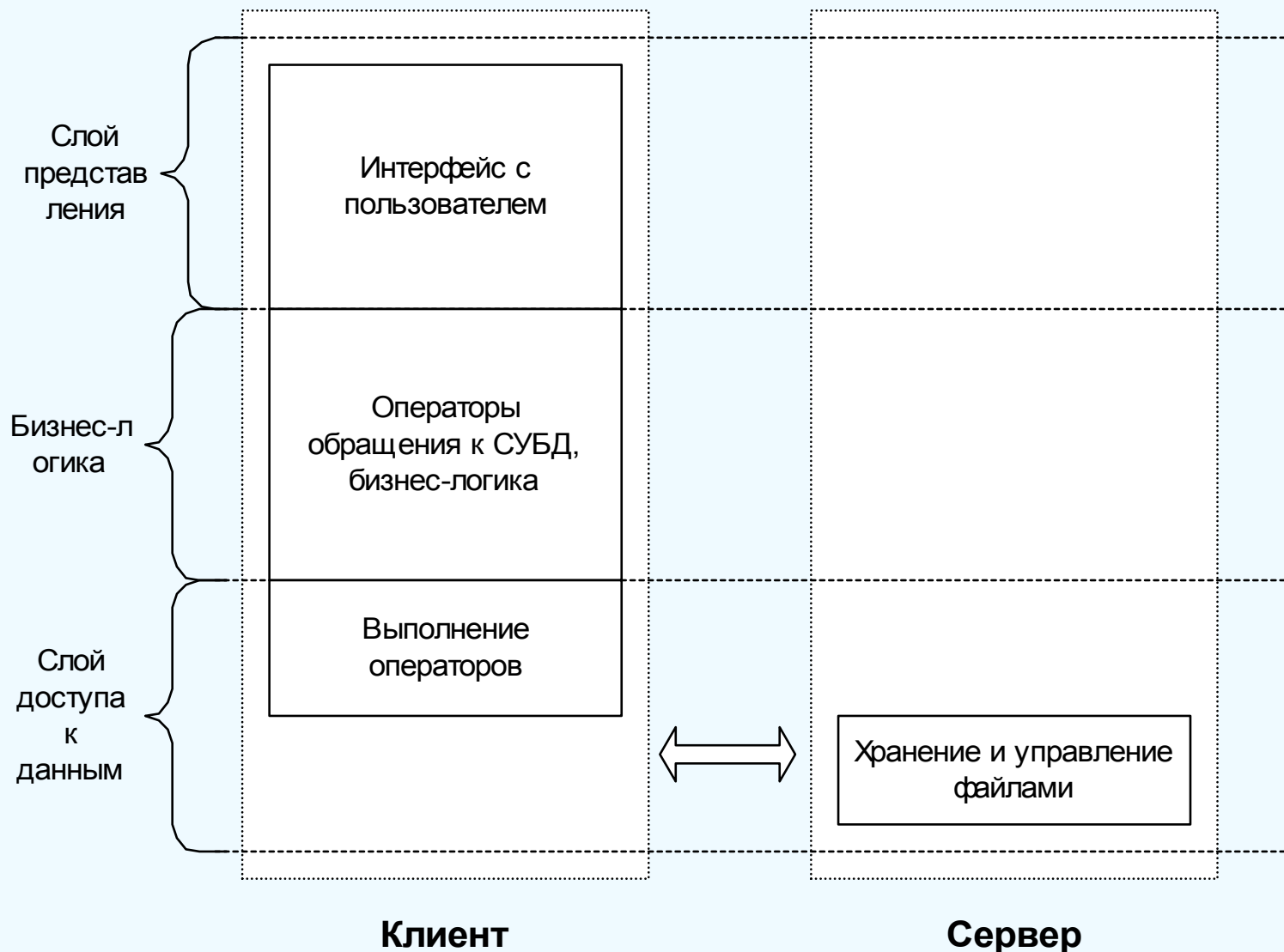
Модели

- файлового сервера;
- сервера СУБД;
- сервера приложений;
- доступ через Intra-/Inter- net CGI/API;
- доступ через Intra-/Inter- net через мигрирующие программы;
- распределенной системы.

Разделение на слои (Gartners Group, и др.)

- представления
- бизнес-логика
- доступа к данным

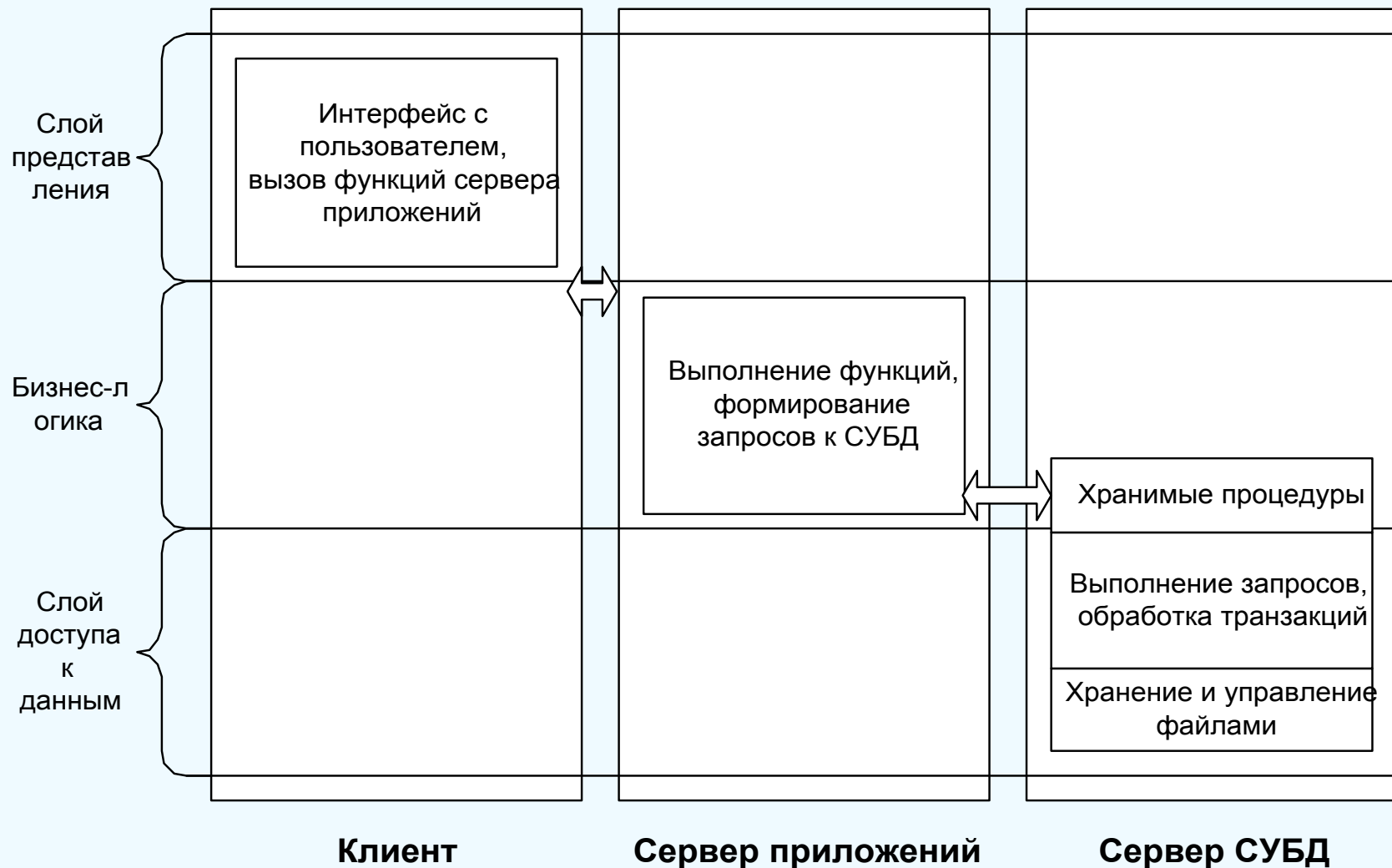
Модель файлового сервера



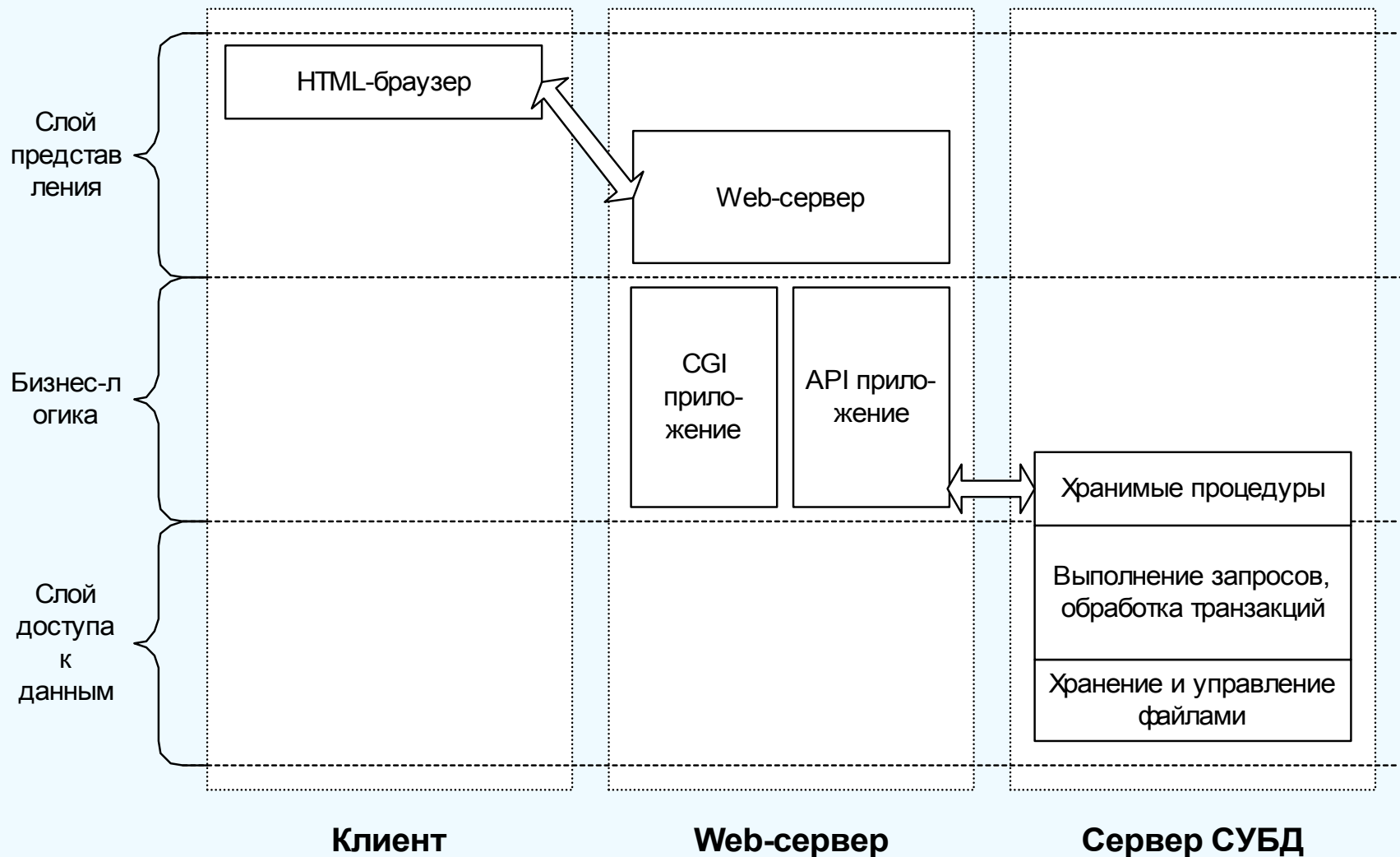
Модель сервера СУБД



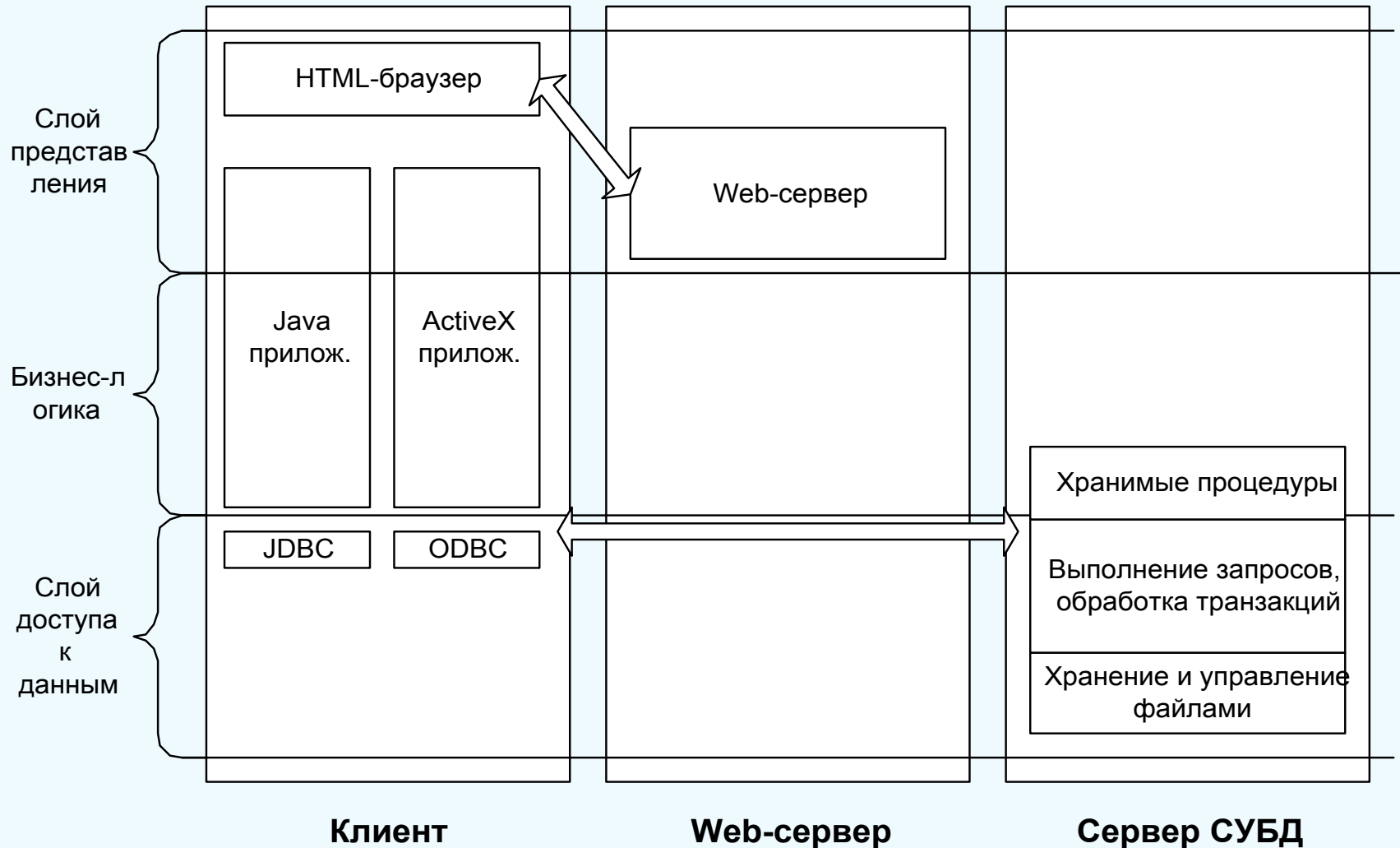
Модель сервера приложений



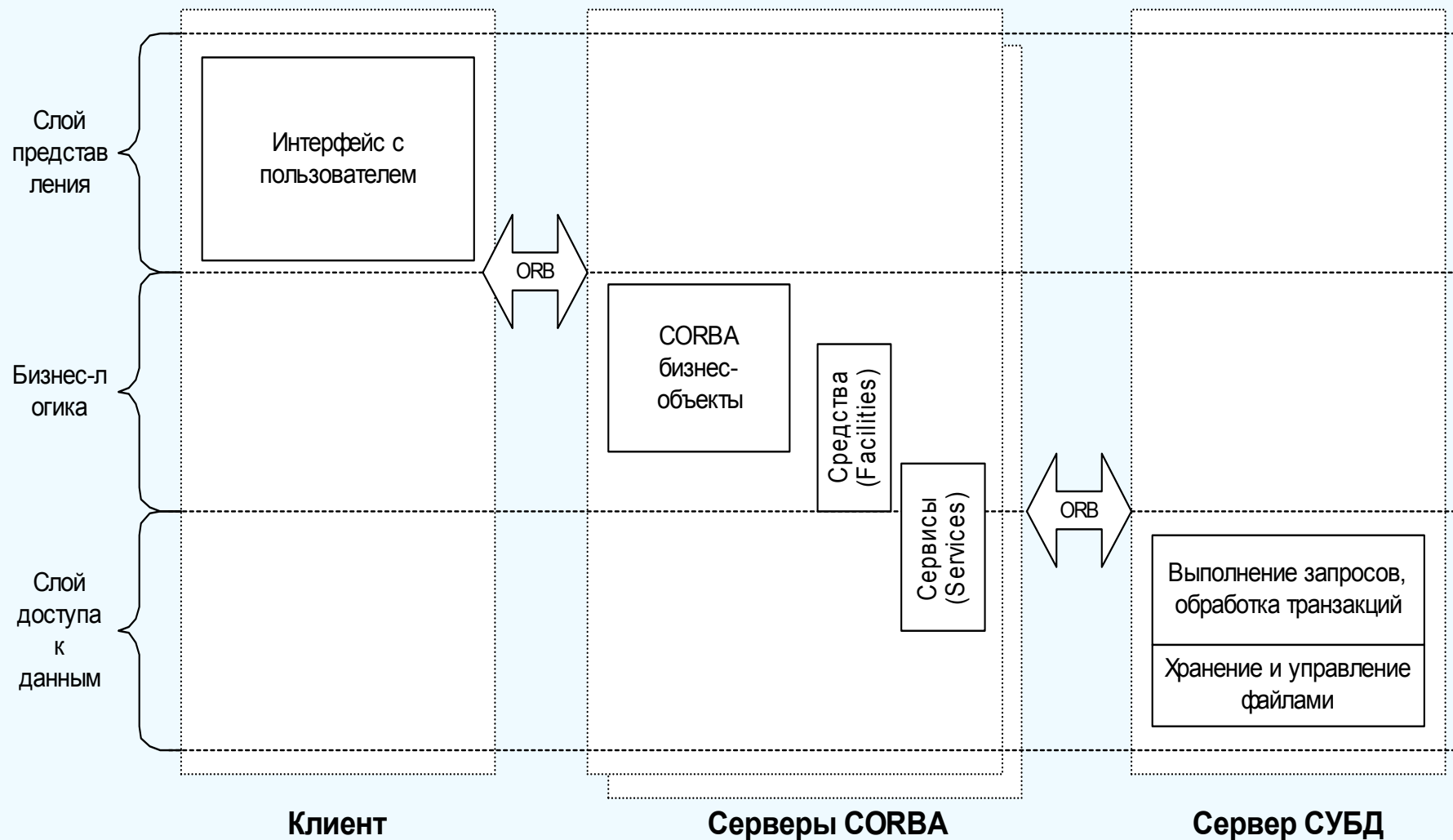
Модель доступа через Intra-/Internet & CGI/API



Модель Inter-/Intra-net с мигрирующими программами



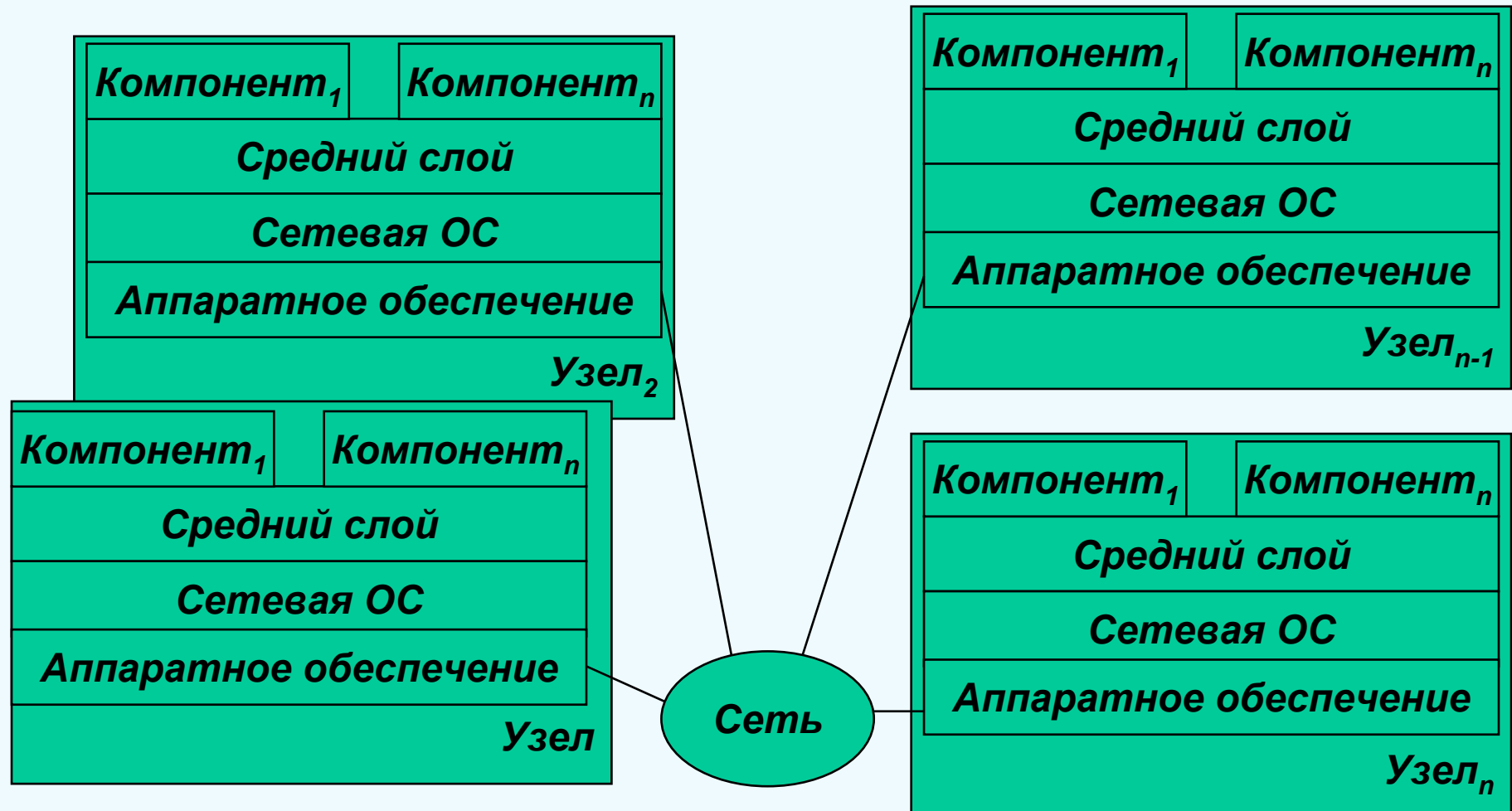
Модель распределенной системы



План лекции

- Архитектура вычислительной системы
- Распределенные системы
- Эволюция объектной технологии
- Литература

Что такое распределенная система?



Распределенная система: определение

Распределенная система - это совокупность автономных узлов, соединенных в вычислительную сеть. Каждый узел выполняет прикладные компоненты и ПО «среднего слоя», позволяющее компонентам координировать свои действия, так что пользователи воспринимают систему как единое целое.

Требования к распределенным системам

- Функциональные
- Нефункциональные (качественные)
 - Масштабируемость (scalability)
 - Открытость (openness)
 - Поддержка неоднородности (heterogeneity)
 - Разделение ресурсов (resource sharing)
 - Отказоустойчивость (fault tolerance)
 - Прозрачность (transparency)

Масштабируемость

- Адаптация распределенных систем к:
 - увеличению числа пользователей
 - уменьшению времени отклика
- Обычно осуществляется увеличением количества обрабатывающих единиц (процессоров, узлов, дисков,)
- Компоненты не должны изменяться при масштабировании
- Архитектура остается неизменной

Открытость

- Вопрос открытости вызван необходимостью изменения, расширения системы
- Детализированные интерфейсы компонентов должны быть опубликованы (документированы)
- Новые компоненты должны быть интегрированы с существующими компонентами (развитие системы, новые функциональные требования)
- Должны быть решены различия в способах представления данных на различных платформах

Неоднородность

- Неоднородность
 - Операционных систем
 - Аппаратных платформ
 - Вычислительных сетей
 - Языков программирования

Разделение ресурсов

- Возможность свободного использования ресурсов в системе (аппаратного, программного обеспечения и данных)
- Менеджер ресурсов контролирует доступ, схему именования и управляет совместным доступом
- Возникает вопрос защиты и разделения доступа

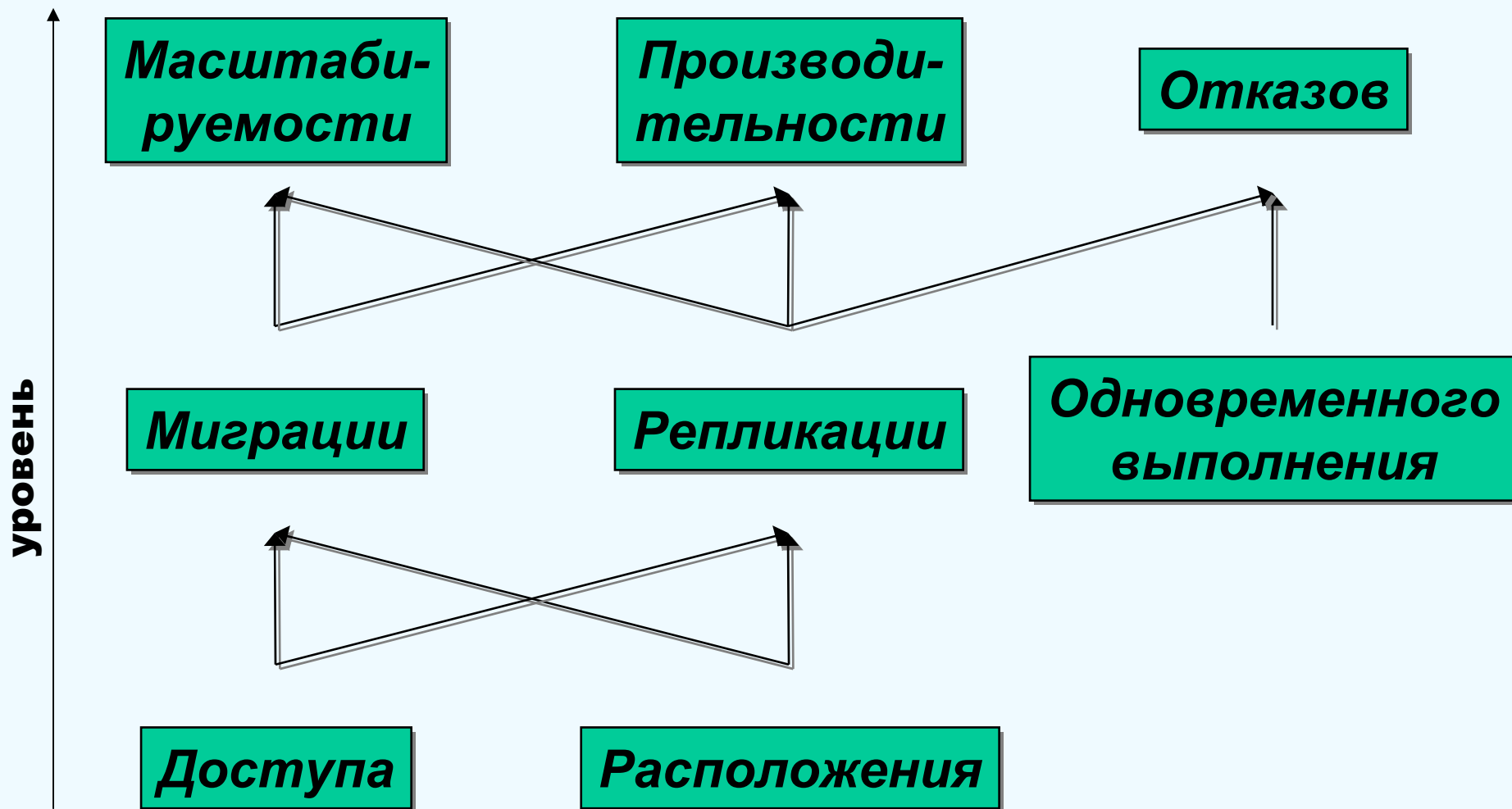
Отказоустойчивость

- Сбои в ПО, аппаратном обеспечении, в СПД происходят!
- Распределенная система должна обеспечивать доступность сервисов даже при низкой надежности компонентов
- Отказоустойчивость достигается
 - Избыточностью (репликация, RAID-5)
 - Восстановлением (журнал БД)

Прозрачность

- Распределенная система должна восприниматься как пользователями, так и программистами как единое целое, а не как простой набор компонентов
- Вопрос прозрачности имеет несколько аспектов (составная часть Международного стандарта по открытой распределенной обработке ISO 20746-2 1996)
- Данные аспекты определяют те свойства, которые должны иметь распределенные системы

Прозрачность



Прозрачность доступа

- Обеспечивает доступ к локальным и удаленным объектам единым образом
- Примеры:
 - Навигация в Web
 - SQL запросы к распределенной БД
 - Файловые операции в NFS

Прозрачность местонахождения

- Обеспечивает доступ к объектам без знания их местоположения
- Примеры:
 - Страницы Web
 - Таблицы в распределенной БД
 - Файлы NFS

Прозрачность одновременного выполнения

- Позволяет независимым процессам получать доступ к разделенным ресурсам одновременно без интерференции и с сохранением целостности
- Пример:
 - NFS
 - Сеть банкоматов
 - СУБД – забота СУБД обеспечить целостность как при единичном, так и одновременном доступе

Прозрачность репликации

- Использование нескольких экземпляров информации для увеличения надежности и производительности. При этом ни для пользователей ни для приложений не известно с чем осуществляется взаимодействие (с копией или оригиналом).
- Пример:
 - Распределенные СУБД
 - Репликация Web-сайтов: round-robin, и др.

Прозрачность отказов

- Скрытие отказов от пользователей, а также от клиентских и серверных компонентов
- Позволяет пользователям и приложениям завершить выполнение операций вне зависимости от сбоев и отказов других компонентов
- Пример:
 - Вызов сервиса в CORBA

Прозрачность миграции

- Возможность перемещения информационных объектов в системе без влияния на работу пользователей или приложений
- Примеры:
 - NFS
 - Страницы Web

Прозрачность производительности

- Динамическая реконфигурация системы в зависимости от изменения нагрузки (балансировка нагрузки)
- Трудно достижима (трудно в априори оценить нагрузку)
- Примеры:
 - СУБД с балансировкой нагрузки

Прозрачность масштабирования

- Позволяет масштабировать систему без изменения системной структуры и алгоритмов приложений
- Примеры:
 - WWW

Литература / Internet ИСТОЧНИКИ

- В. Эммерих *Конструирование распределенных объектов*. - М.: Мир. - 2002.
- М.Р. Когаловский *Энциклопедия технологий баз данных*. - М.: ФС. - 2002.
- Ю.А. Григорьев, А.Д. Плутенко. - *Жизненный цикл проектирования распределенных баз данных*. - Благовещенск. - 1999.
- www.omg.org