

Тема 2

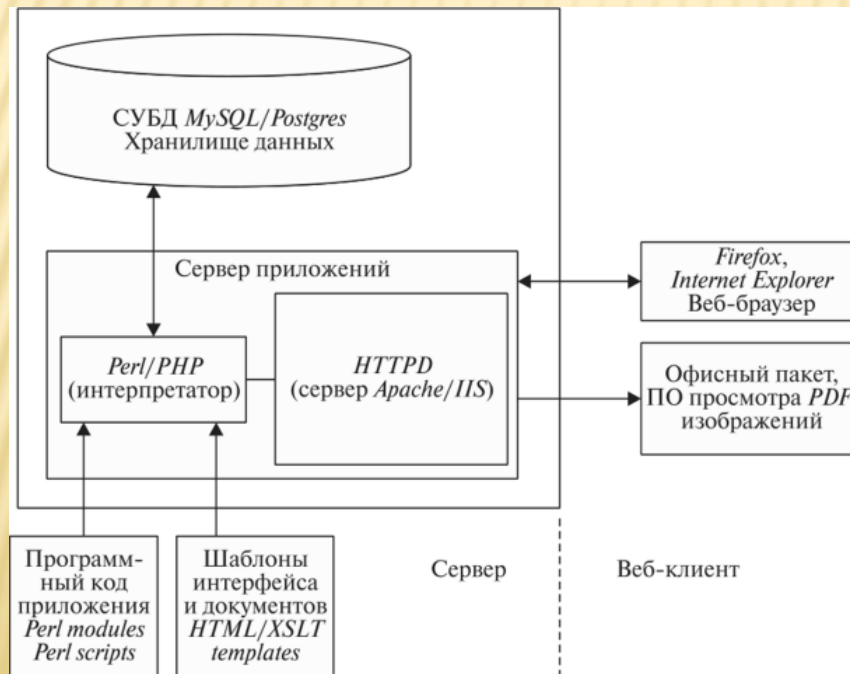
АРХИТЕКТУРЫ РАСПРЕДЕЛЕННЫХ ИНФОРМАЦИОННЫХ СИСТЕМ

ПОНЯТИЕ ОРГАНИЗАЦИИ РАСПРЕДЕЛЕННОЙ СИСТЕМЫ

- Организация РС определяется тем каким образом программное обеспечение РС распределяется между вычислительными узлами этой системы.
- В организации систем часто выделяют:
 - Логическую организацию совокупности программных компонент системы;
 - Физическую организацию размещение этих компонент на узлах системы.

ПРОГРАММНАЯ ОРГАНИЗАЦИЯ

- Организация РС определяется составом программных компонент входящих в состав системы.
- Программная организация показывает из каких программных компонентов состоит система, а также и то как взаимодействуют между собой программные компоненты этой системы.



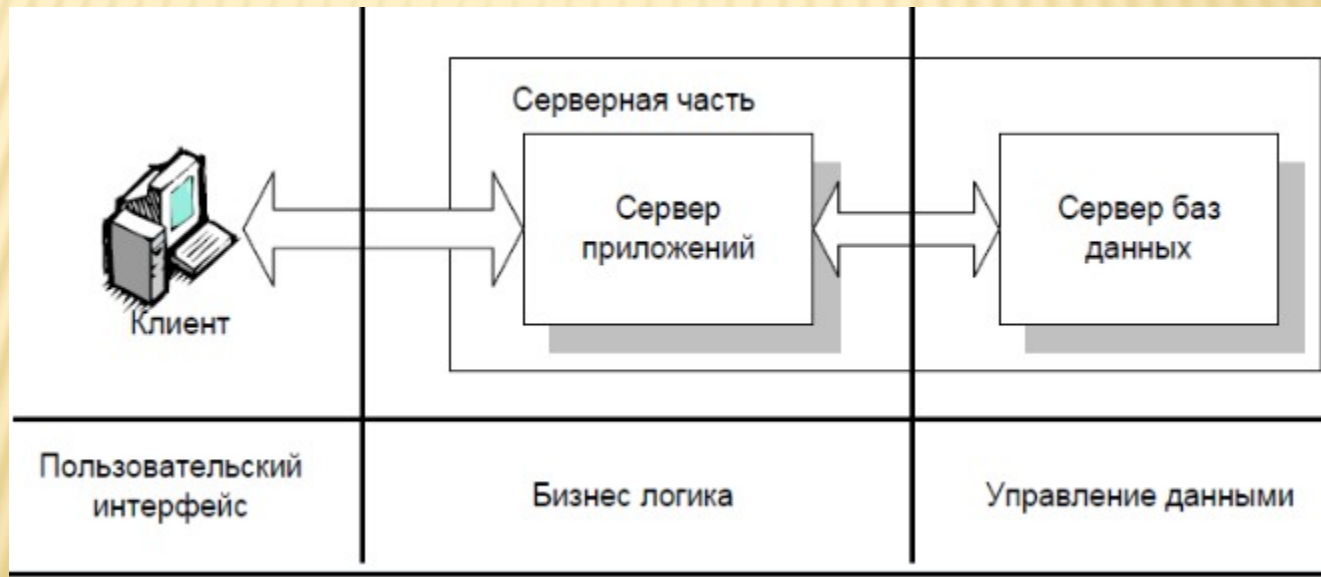
ПРОЗАЧНОСТЬ РИС И ЕЕ АРХИТЕКТУРА

- Исходя из требования обеспечения прозрачности в распределенных системах требуется четко разделять приложения и лежащие в их основе платформы.
- Такое разделение в РС выполняется с помощью промежуточного уровня системы.

Распределенное приложение
ПО промежуточного слоя (Middleware Layer)
Аппаратно-программная платформа

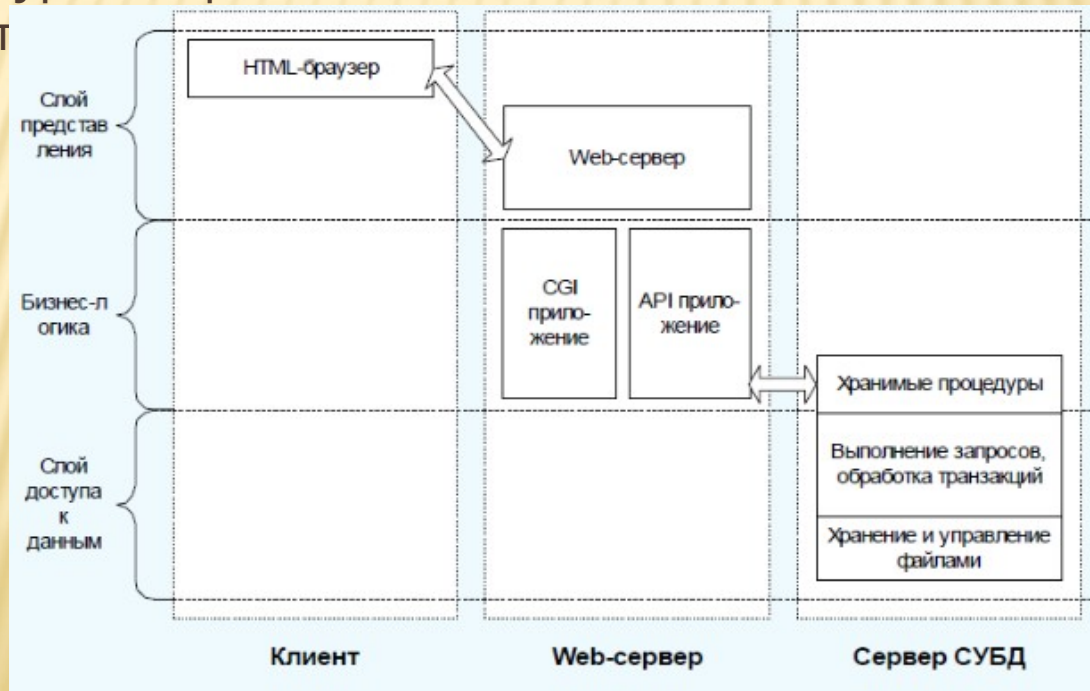
ВЫБОР ВАРИАНТА ПРОГРАММНОЙ АРХИТЕКТУРЫ

- Важнейшим решением при разработке архитектуры системы является:
 - выбор варианта размещения ПО промежуточного уровня (ППУ) системы.
- ▣ Имеется различные методики определение состава и размещения ППУ приложений, что и определяет множество вариантов программных архитектур.



СИСТЕМНАЯ АРХИТЕКТУРА

- Фактическая (реально разворачиваемая) реализация РС, требует однозначного определения размещения программных компонент системы на реальных машинах.
- Практически всегда имеется множество вариантов такого размещения.
- Размещение программных компонент системы (программная архитектура) на физических машинах называется системной архитектурой.



ВИДЫ СИСТЕМНОЙ АРХИТЕКТУРЫ

- Различают три вида системной архитектуры:
 - централизованная;
 - децентрализованная (peer-to-peer);
 - гибридная – комбинация элементов централизованной и децентрализованной архитектур.

ПОНЯТИЕ АРХИТЕКТУРНОГО СТИЛЯ

- В настоящее время исследования в области программного обеспечения достигли достаточной зрелости, что позволило однозначно определить понятие архитектурного стиля (архитектуры) РИС.
- При проектировании и создании РС выбор архитектуры является ключевым техническим решением, определяющим успех создания больших программных систем.
- При обсуждении архитектурных аспектов РС важным понятием является архитектурный стиль, который описывается в терминологии компонент и определяет:
 - способ коммуникаций между компонентами;
 - порядок обмена данными между компонентами;
 - как элементы системы совместно формируют распределенную систему.

ПОНЯТИЕ ПРОГРАММНОГО КОМПОНЕНТА

- **Компонент** – модульная единица ПО снабженная полностью определенным и предоставляемым по запросу интерфейсом.
- Компонент должен обладать свойством – **заменяемости** (*replaceable*) в рамках системного окружения. Замена компонента может быть выполнена в любой момент, даже в условиях работы системы. Последний аспект определяет, что в РС может отсутствовать опция
- **Интерфейс** - описывает состав параметров необходимых для обращению к компоненту. Замена компонента может быть выполнена только при условии неизменности его интерфейса.
- **Конектор** – это механизм который обеспечивает коммуникации, и способствует координации (или кооперации) компонент друг с другом.
- Конектор может быть сформирован на основе средств реализующих способ связи между компонентами. :
 - удаленный вызов процедур (RPC);
 - обмен сообщениями (message passing);
 - потока данных (streaming data) и д.р.
- Использование понятий компонент и конектор позволяет описывать различные варианты конфигураций, которые в свою очередь могут быть квалифицированы как **архитектурные стили**.

ОСНОВНЫЕ ВИДЫ АРХИТЕКТУР РИС

- В настоящее время общепризнанными архитектурными решениями (стилями) считаются:
 - многоуровневые архитектуры (layered);
 - объектные архитектуры (object-based);
 - компонентные архитектуры (component-based);
 - сервисно-ориентированные архитектуры;
 - ресурсо-центрированные архитектуры (resource-centered);
 - архитектуры основанные на событиях (event-based).
- Эти архитектурные решения могут одновременно применяться в одних и тех же

АРХИТЕКТУРНЫЕ МОДЕЛИ РАСПРЕДЕЛЕННЫХ СИСТЕМ

АРХИТЕКТУРА СИСТЕМЫ

- Архитектура системы - это ее структура в терминах отдельно определяемых компонент (сущностей) и взаимоотношение между ними.
- Главной целью выбора возможной архитектуры РИС является обеспечение следующих характеристик системы:
 - масштабируемости;
 - надежности;
 - открытости;
 - управляемости;
 - гибкости;
 - адаптируемости;
 - эффективности;
 - безопасности;
 - и т.д.

ХАРАКТЕРИСТИКИ СИСТЕМЫ

МАСШТАБИРУЕМОСТЬ

Масштабируемость – сохранение работоспособности системы при изменении масштаба системы.

- Масштабируемость системы может измеряться по трем показателям:
 - По размеру.
 - По географической распределенности.
 - По степени сохранения управляемости при увеличении масштаба РИС

МАСШТАБИРОВАНИЕ ПО РАЗМЕРУ

Наиболее часто проблемы размера определяются заложенными при ее создании централизацией:

- служб;
- данных;
- алгоритмов.

Концепция	Пример
Централизованная служба	Один сервер на всех пользователей
Централизованные данные	Единый телефонный справочник на всех пользователей доступный только в режиме подключения
Централизованные алгоритмы	Организация системы имен Интернет на основе единого файла хостов сети

МАСШТАБИРУЕМОСТЬ И ПРОИЗВОДИТЕЛЬНОСТЬ

- Вертикальное масштабирование – увеличение производительности каждого компонента системы с целью повышения общей производительности.
 - –возможность заменять в существующей вычислительной системе компоненты более мощными.
- Горизонтальное масштабирование – возможность добавлять к системе новые узлы (серверы, процессоры) для увеличения общей производительности.

НАДЕЖНОСТЬ

- Надежность охватывает множество требований к распределенным системам, из которых важнейшими являются:
 - доступность(availability);
 - безотказность(reliability);
 - безопасность(safety);
 - ремонтпригодность(maintainability).

ДОСТУПНОСТЬ (ГОТОВНОСТЬ)

- ▣ Доступность - свойство системы находиться в состоянии готовности к работе.
- ▣ Обычно доступность – это вероятность того, что система в данный момент времени будет правильно работать и окажется в состоянии выполнить свои функции, если пользователи того потребуют.
- ▣ Система с высокой степенью доступности – это такая система, которая в произвольный момент времени, скорее всего, находится в работоспособном состоянии.

КЛАССИФИКАЦИЯ СИСТЕМ ПО ПОКАЗАТЕЛЮ ДОСТУПНОСТИ (ГОТОВНОСТИ)

Класс готовности	Показатель готовности %	Продолжительность простоев в год	Степень надежности
1	90	более 1 месяца	надежная система
2	99	менее 4 дней	надежная система
3	99,9	менее 9 часов	высоконадежная система
4	99,99	около 1 часа	отказоустойчивая система
5	99,999	около 5 минут	безотказная система
5	99,9999	около 30 секунд	безотказная

ОТКРЫТОСТЬ РИС

- Обеспечивает:
 - интероперабельность - способность к взаимодействию с другими РИС
 - переносимость – способность к переносу приложений между разными РИС.
- Основывается на использовании при проектировании ИС открытых стандартов, спецификаций и протоколов, определяющих порядок взаимодействия между всех компонентов распределенной системы между собой, а также пользователей с распределенной системой.
- Реализует разделение правил выполнения взаимодействий от механизмов их реализующих.

ГИБКОСТЬ РИС: ВИДЫ

- Гибкость может быть следующих видов:
 - Программной
 - Замена программ (например, выпуск новых версий) и перенос их между хостами РИС
 - Аппаратной
 - Изменение числа и характеристик технических средств
- Структурной
 - Изменение структуры РИС (например, переподчинение элементов, разбиение одной РИС на несколько, слияние нескольких РИС в одну)
 - Логической
 - Изменение бизнес процессов (например, последовательности операций)

БЕЗОПАСНОСТЬ РИС

- Безопасность определяет, насколько катастрофична ситуация временной неспособности системы должным образом выполнять свою работу.
- По-настоящему безопасную систему построить крайне тяжело.

АРХИТЕКТУРНЫЕ ЭЛЕМЕНТЫ

АРХИТЕКТУРНЫЕ ЭЛЕМЕНТЫ

- Для понимания того, что является строительными блоками распределенных систем, необходимо решить четыре ключевых вопроса:
 - 1. Какие сущности взаимодействуют в рамках распределенной системы?
 - 2. Каким образом эти сущности взаимодействуют друг с другом? Или более точно: какую коммуникационную модель (парадигму) они используют?
 - 3. Какие роли играют (выполняемые ими функции), и какую ответственность они несут в рамках всей системы?
 - 4. Каким образом сущности системы (логические) отображаются на физические элементы распределенной инфраструктуры?

СУЩНОСТИ РИС

- АРХИТЕКТУРА РИС определяет:
 - какие сущности входят в РС,
 - каким образом они взаимодействуют.
- Выбор сущностей и способы их взаимодействия в рамках РИС определяет весь спектр возможных архитектурных решений.
- Сущности, образующие РИС, имеют два аспекта:
 - системный;
 - проблемный.

СИСТЕМНАЯ РЕАЛИЗАЦИЯ СУЩНОСТИ

- Системная сторона сущности определяет способ **физической реализации функций сущности** и методов ее **взаимодействия** с другими сущностями.
- В современных ИС способ такой физической реализации является **процесс (задача) ОС**, выполняемый на аппаратных средствах физического узла и получает для своего исполнения все необходимые ресурсы (вычислительные, ОП, устройства хранения (SDD, HDD), ВУ, сетевые устройства).
- Исходя из системной стороны сущностей, взаимодействие между ними рассматривается как модель **межпроцессного** взаимодействия между распределенными сущностями.
- При этом помимо классического процесса имеются следующие варианты их реализации:
 - В примитивных системах, например, таких как системы датчиков, ОС узлов системы могут не поддерживать механизм процессов, в этом случае такие сущности называют **узлами**;
 - В большинстве современных ОС абстракция процесса дополняется абстракцией **потока**. Потоки, как и процессы могут обслуживать **конечные точки подключений**.
- Точка подключения: - сетевой адрес узла/процесса/потока. Пример:
UDP / TCP

ПРОБЛЕМНАЯ СТОРОНА СУЩНОСТИ

- Определяется предметной областью, для автоматизации которой используются сущности РС. Предметная сторона определяет:
 - алгоритмы обработки информации;
 - порядок обмена информацией между сущностями в рамках РС.
- Наиболее часто в качестве проблемно-ориентированной абстракции сущностей РИС используется:
 - объекты;
 - компоненты;
 - службы;
 - ресурсы.

Сущности РИС	
Системно-ориентированные сущности	Проблемно-ориентированные сущности
Узлы	Объекты
Процессы/потоки	Компоненты
	Сервисы (Web сервисы)
	Ресурсы

ОБЪЕКТЫ

- ▮ Выбор объектов в качестве сущностей РИС обусловлен во многом объектно-ориентированным подходом в программировании, а также к проектированию РИС.
- ▮ Объекты РС представляют собой абстракцию объектов реального мира, относящихся к конкретной проблемной области.
- ▮ Разделение предметной области на объекты выполняется исходя из естественной декомпозиции предметной области на ряд взаимодействующих друг с другом объектов.
- ▮ Доступ к объектам реализуется через их интерфейсы, которые описываются с помощью языка IDL-Interface Definition Language, обеспечивающего описание методов, определенных для объекта.
- ▮ Распределенные объекты наиболее часто используются в качестве проблемных сущностей в распределенных системах.

ОБЪЕКТНЫЕ КОМПОНЕНТЫ

- Использование объектов в качестве сущностей РС привело к появлению ряда проблем, ответом на которые явилось использование компонент. Компоненты сходны с объектами. Они также как объекты:
 - являются абстракциями прикладной области;
 - доступны через интерфейсы.
- Ключевым отличием компонент от объектов состоит в том, что они не только предоставляют интерфейсы, но также содержат информацию о том, какие компоненты/интерфейсы должны быть представлены для компонента, чтобы он мог выполнять свои функции.
- Иными словами, явно разрешаются все зависимости, и обеспечивается полное выполнение всех правил, принятых при конструировании системы, что позволяет разработку компонент третьими сторонами и способствует при реализации проекта более полной и адекватной реализации.

СЕРВИСЫ

- ▣ Сервисы представляют третью парадигму определения сущностей, используемую при создании РС.
- ▣ Сервисы являются сущностями близкими к объектам и компонентам, также использующими подход, основанный на инкапсуляции поведения в компонент и использовании интерфейса для получения доступа друг к другу.

WEB-СЕРВИСЫ

- В отличие от первых двух видов сущностей web-сервисы в действительности интегрированы в web и используют стандарты www для представления и описания сервисов.
- W3C определил Web-сервисы как:
 - Программное приложение, идентифицируемое с помощью URI, чей интерфейс и связывание реализуется на основе использования XML.
 - Web-сервис поддерживает прямое взаимодействие с другими программными агентами, используя обмен XML-сообщениями по Internet протоколам.
- Если объекты и компоненты часто используются внутри организаций для разработки тесно связанных приложений, то Web-сервисы в целом выглядят как полноценные сервисы, которые могут при их комбинировании образовывать сложные сервисы с дополнительными возможностями.
- Web-сервисы часто используются для интеграции B2B, что требует от них преодоления границ, разделяющих организации. Web-интерфейсы могут быть реализованы разными провайдерами и использовать различные низкоуровневые технологии.

РЕСУРСЫ

- Рост числа сервисов доступных через Web и создание распределенных систем на основе композиций сервисов привели к необходимости переосмысления архитектуры РИС построенных на базе Web.
- Одной из проблем построения РИС на основанных на Web сервисах явилась высокая сложность обеспечения связи между большим числом различных компонент.
- В качестве альтернативы было предложено рассматривать РИС как коллекцию ресурсов каждый из которых индивидуально управляется своим компонентом (сервисом).

СПАСИБО ЗА ВНИМАНИЕ !