

### Java:

технология Enterprise Java Beans 3.0

Лекция 1. Обзор технологии Java2EE



Лектор: Надеинский Лев Андреевич

e-mail: n.lev.a@ya.ru

skype: L\_E\_V-



·Математик по специальности компьютерная безопасность.

#### Проекты:

Денежные переводы "Блиц" (СберБанк)

Автоматизированная система "Дивиденды" (СберБанк)

Автоматизированная система "Выплаты" (СберБанк)

Система автоматизированного сбора данных (Infoskidka.ru)

### Цели лекции

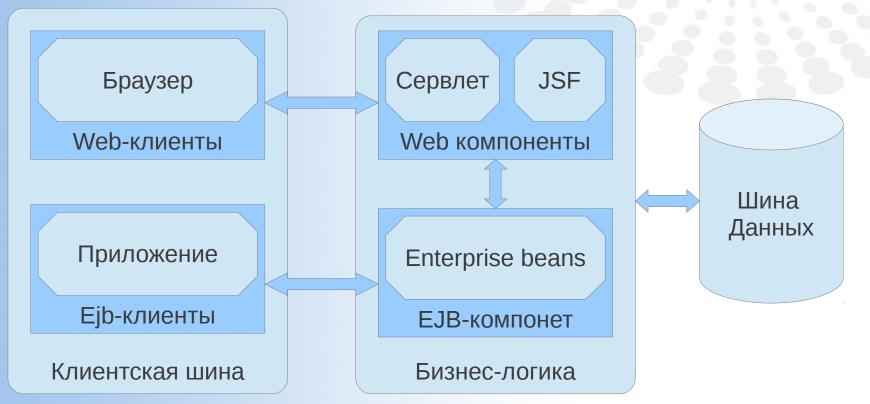
- По окончинию занятия Вы должны ознакомиться с:
  - Описанием платформы Java™ Enterprise Edition (Java EE)
  - Различными компонентами Java EE
  - Возможными вариантами развертывания Java EE приложений

## Платформа Java Enterprise Edition

- Java Enterprise Edition (Java EE) платформа является стандартом для разработки и реализации корпоративных приложений
  - Платформа предоставляет многоуровневую архитектуру для построения приложений
  - Помогает улучшить процесс разработки, развертывания и реализации корпоративную приложений. Охватывает весь цикл жизни приложения
  - Позволяет подключать большое количество разработчиков в рамках одного приложения

### Платформа Java Enterprise Edition

- Многоуровневые распределенные приложения
- Поддержка компонентной модели построения приложений



Страница ¬ 5 www.specialist.ru

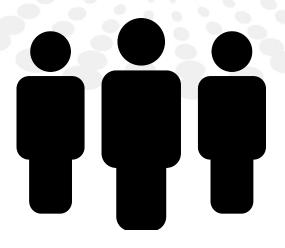
## **Достоинсива платформы Java EE**

- "Пришем один раз, используем повсеместно". Предоставляет простую компонентную разработку.
- Дает возможность выбора продуктов и вендоров, которые поддерживают платформу Java EE, что предоставляет широкий выбор при развертывании приложения.
- Возможна интерграция с существующими приложениями по API этих приложений
- Разделяет потребности клиента и бизнес-логику
- Предоставляет множество сценариев проектирования и разработки
- Позволяет различным клиентам использовать одну бизнес-логику

www.specialist.ru

## **Достоинсива платформы Java EE**

- Платформа предоставляет возможность легко разделять по вертикали обязанности программистов в рамках одного проекта по конкретным направлениям:
  - Создание web-компонент
    - JSF, GWT
    - Сервлеты
    - Валидаторы
  - Бизнес-логика и правила работы приложения
  - Развертывание и внедрение



# Развитие платформы

	Описание	Версия		
		J2EE	JEE 5	JEE 6
EJB	Enterprise JavaBeans —	2.1	3.0	3.1
JPA	Java Persistence API	Нет	1.0	2.0
Сервлет	Обслуживание запросов веб-клиентов.	2.4	2.5	3.0
JSP	JavaServer Pages — Динамическая генерация веб-страниц на стороне сервера.	2.0	2.1	2.2
JSTL	JavaServer Pages Standard Tag Library	Нет	1.2	1.2
JSF	JavaServer Faces	Нет	1.2	2.0
JMS	Java Message Service — обмен сообщениями.	1.1	1.1	1.1
StAX	Streaming API for XML	Нет	1.0	1.0
CDI	Context and Dependency Injection	Нет	Нет	1.0

Страница ¬ 8 www.specialist.ru

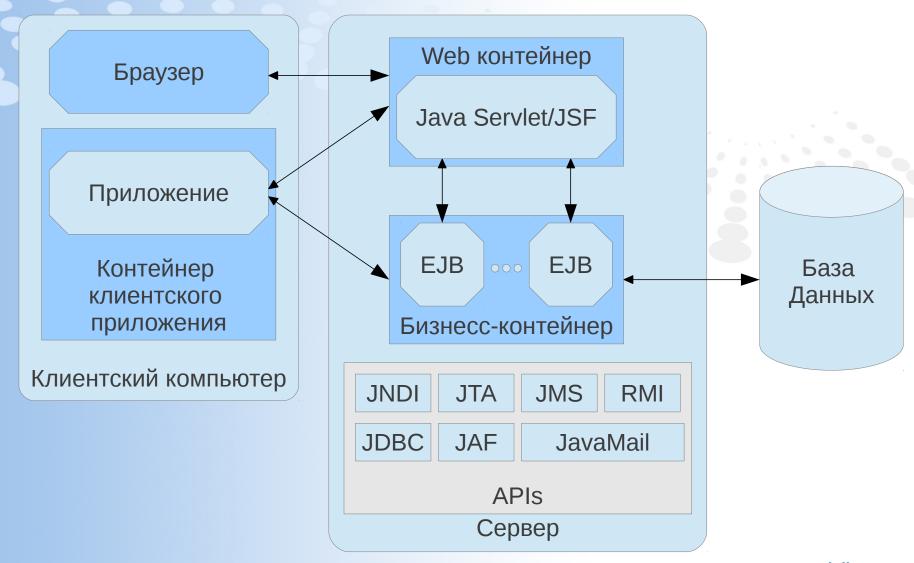
#### Компоненты

• В спецификации Java EE перечислены следующие компоненты:

- Java Servlet
- JavaServer Pages
- Enterprise JavaBeans 3.0
- JDBC 3.0
- RMI-IIOP
- JNDI 2
- Web services

- Java Message Service
- Java Transaction API
- Java Authentication and Authorization Service
- Java EE Connector Architecture
- SOAP with Attachments API for Java

## Архитектура Java EE



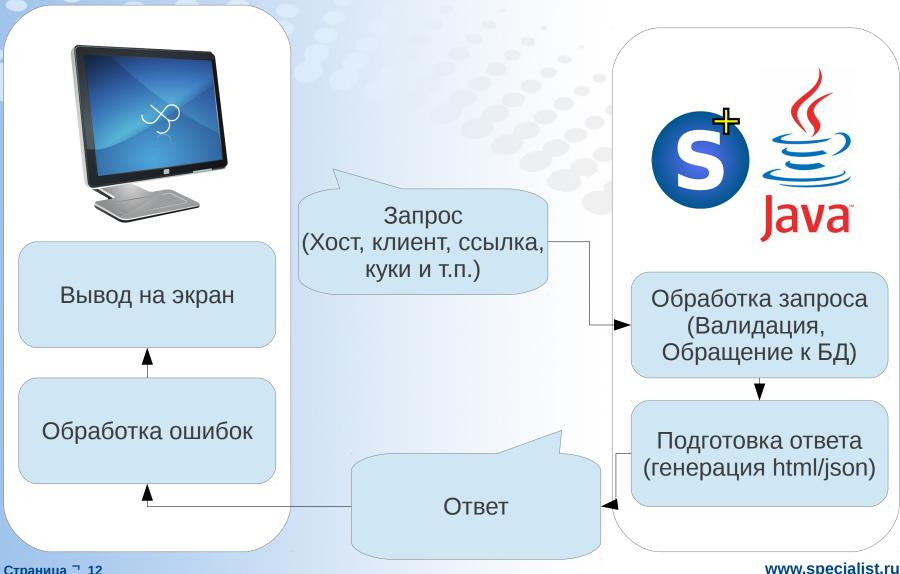
Страница ¬ 10 www.specialist.ru

### Компоненты веб-шины

- Веб-уровня может состоять из:
  - Јаvа сервлеты
  - JSPs
- Servlets и JSP-страниц:
  - Работа на запрос-ответ моделью
  - Создание динамического HTML
  - Доступ к базе данных через JDBC
  - Доступ к бизнес-уровня компонентов
  - Обработака клиентских событий
  - Генерация визуальных интерфейсов, таких как веб-страницы

Страница ¬ 11 www.specialist.ru

## Сервлет



www.specialist.ru Страница 7 12

# JavaServer Page (JSP)

- Веб-уровня может состоять из:
  - Јаvа сервлеты
  - JSPs
- Servlets и JSP-страниц:
  - Работа с моделью на запрос-ответ
  - Создание динамического HTML
  - Доступ к базе данных через JDBC
  - Доступ к бизнес-уровня компонентов
  - Обработака клиентских событий
  - Генерация визуальных интерфейсов, таких как веб-страницы



## JavaServer Faces (JSF)

#### JSF:

- Это серверный компонент для веб-приложений
- Реализует Model-View-Controller (MVC)
- Обеспечивает разделение навигационных и потоков данных
- Создан для разработки стиле RAD

### Web шина

• Web-компоненты генерируют динамический контент.

#### Servlets:

- Расширение функциональности веб-сервера
- Лучше всего использовать для бизнес-логики

#### • JSP-страницы:

- Комбинация HTML (или другой разметки) и Java
- Лучше всего использовать для логики представления

#### JSFs:

- Компонентная основа
- Реализация МVС

#### Компоненты бизнес-шины

#### • Бизнес-компоненты:

- EJBs
- бизнес-логика
- Получение данных от клиентских программ
- Получение данных из базы данных хранилища
- Обработка данных и взаимодействовать с базой данных и программой-клиентом
- Может быть вызван Web-компонентами

## **Enterprise JavaBeans (EJB)**

### • EJB:

- Это серверные компоненты, написанные на Java
- Описывают бизнес-логику приложений предприятия
- Размещаются в контейнерах EJB
- Основаны на Remote Method Invocation (RMI) связях
- Платформо-независимы
- Предоставляют удаленные сервисы для клиентов
- Могут быть представлены в виде веб-сервисов
- Используют JDBC для подключения к базе данных

## Коммуникация Java EE через API

- Общение в Java EE реализуется через интерфейсы API, которые включают:
  - RMI
  - JNDI
  - JDBC
- Эти API облегчают связи:
  - Между компонентов Java EE
  - С базой данных

### Коммуникация Java EE через API

#### • Сервисы предоставляемые системой:

- Нейминг и службы каталогов (JNDI Java Naming and Directory Interface)
- Безопасные сервисы для веб-компонентов и компонентов EJB (JAAS)
- Транзакционная архитектура (JTA)
- Сервис обмена сообщениями (JMS)
- Сервис электронной почты (JavaMail)

#### • Удаленные подключения:

- Предприятие бобы (RMI / IIOP, ORMI)
- Servlet / JSP (HTTP, HTTPS, FTP)

Страница ¬ 19 www.specialist.ru

### Java EE приложения

- Java EE приложения состоят из компонентов Java EE и размещены в виде модулей:
  - Веб-модули содержат пользовательский интерфейс:
    - HTML, JSP и сервлеты.
  - ЕЈВ модули содержат компоненты ЕЈВ.
  - Клиентские модули обеспечивают доступ к удаленным приложениям.
  - Информация о пакетах определяет зависимости между модулями.

#### Итоги

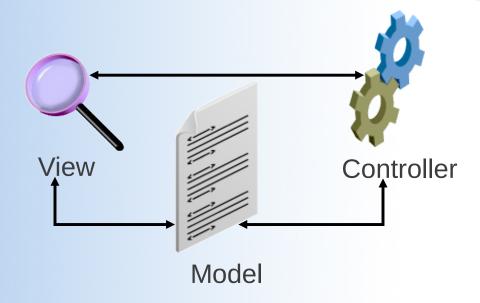
- В этом уроке, вы должны узнали, что:
  - Java EE представляет собой набор технологий, которые поддерживают непрерывную разработку приложений
  - Компоненты являются основой Java EE архитектуры
  - **Веб-компоненты** (сервлеты, JSP, и JSFs) <mark>генери</mark>руют динамический контент
  - **Бизнес-компоненты** (EJB) являются серверными компонентами, которые содержат бизнес-логику

## Шаблоны проектирования

- Повторяющиеся вопросы при разработке приложений привели к внедрению шаблонов проектирования. Общепринятые шаблоны проектирования включают (но не ограничиваются) следующее:
  - Уровень представления:
    - Intercepting filter
    - Сервлет-контроллер (как в MVC)
  - Уровень бизнес-логики:
    - Data Access Object
    - Data Transfer Object

### **Model View Controller**

- MVC отдельно определяет компоненты приложения, как:
  - Бизнес-функциональность (model)
  - Представление (View)
  - Логика управления (Controller)



Страница ¬ 23 www.specialist.ru

### Model

- Модель представляет корпоративные данных и бизнес-правила, обрабатывает доступ и обновления.
  - Вы можете упростить модель с помощью двух механизмов называемый "facade class" и "command pattern".
    - Facade инкапсулирует и координирует действия между сотрудничающими классов.
    - Command pattern инкапсулирует каждую функцию приложения в отдельный класс.
  - Эта модель часто реализуется как EJB.



Страница ¬ 24 www.specialist.ru

### **View**

- View нацелен на презентацию и отвечает за поддержание соответствия между отображаемыми данными и моделями изменений. Это позволяет:
  - Изменять презентацию без изменения логики приложения
  - Разрабывать веб-страницы сотрудниками, имеющими только навыки проектирования UI
- View обычно реализованы в виде JSP.

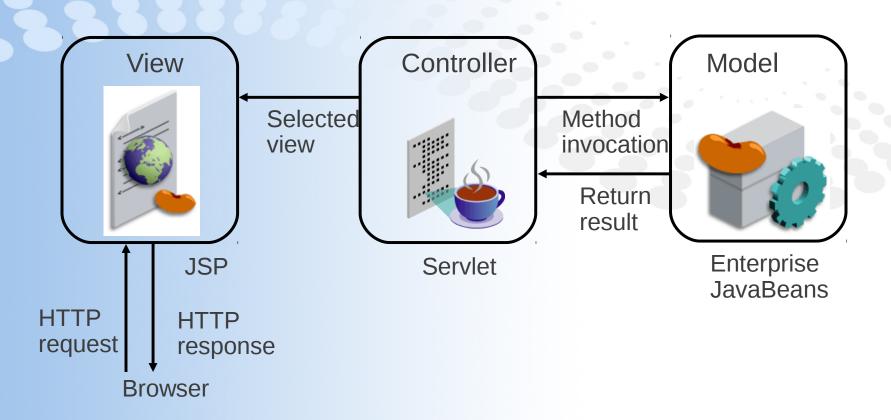


### Controller

- **Контроллер** обеспечивает взаимодействие с клиентом, объединяет модель с представлением.
- Контроллер:
  - Интерпретирует запросы пользователя и управляет бизнес-объектами для выполнения этих запросов
  - Удаляет навигации кодирования с точки зрения
  - Может быть реализован в клиент, Web или EJB уровня, или в комбинации этих уровней
- Контроллер обычно реализуется как сервлет.
- Struts и JSF две широко используемые реализации.



# Пример MVC



Страница ¬ 27 www.specialist.ru



Страница ¬ 28 www.specialist.ru

### Домашнее задание

По вопросам высылки домашних заданий обращайтесь по email:

N.LEV.A@ya.ru