

Sistemas Distribuídos na WEB

(Plataformas para Aplicações Distribuídas)

J2EE (Java 2 Enterprise Edition)

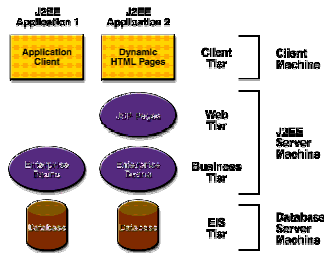
Sumário

- ✳ Introdução J2EE (Java 2 Enterprise Edition)
- ✳ Arquitetura J2EE
- ✳ APIs J2EE
- ✳ Web Container: Servlets e JSP
- ✳ Padrão XML

Java 2 Enterprise Edition

- ✳ J2EE é
 - Uma especificação para servidores de aplicação que define padrão de suporte a componentes e serviços
 - Um pacote de APIs e ferramentas para desenvolver componentes que rodam nesses servidores
- ✳ É objetivo da plataforma J2EE reduzir o custo e a complexidade de desenvolver serviços multi-camada
- ✳ Servidores de aplicação compatíveis com a especificação J2EE oferecem
 - Suporte à arquitetura de componentes EJB
 - Suporte a serviços Web, Servlets e JSP

Java 2 Enterprise Edition

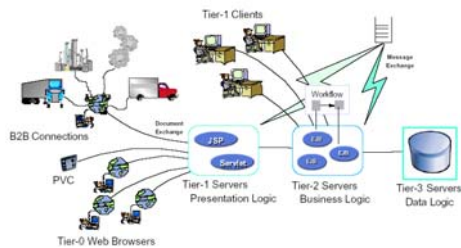


Multitiered Applications

Sistemas Distribuidos 2007
Prof. Carlos Paes

4

Java 2 Enterprise Edition

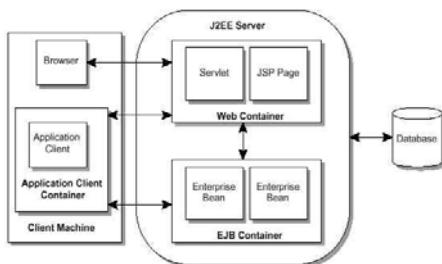


Multitiered Applications Environments

Sistemas Distribuidos 2007
Prof. Carlos Paes

5

Java 2 Enterprise Edition



Sistemas Distribuidos 2007
Prof. Carlos Paes

6

Java 2 Enterprise Edition (Camadas)

Componentes J2EE na Camada de Apresentação

- Os seguintes componentes podem existir na camada de apresentação:
- "Application client" (cliente não-Web)
 - Tipicamente usa Swing como User Interface (UI)
 - Também chamado "Console Application"
- Applets

Java 2 Enterprise Edition (Camadas)

Componentes J2EE na Camada Web

- Componentes da camada Web podem incluir vários módulos, incluindo:
- Páginas HTML/XML estáticas
 - Servlets
 - Programas em Java que rodam no servidor Web e que processam pedidos gerando respostas dinâmicas
- Java Server pages (JSP)
 - Templates HTML mais fáceis de criar, mas contendo "scriptlets" (trechos em Java) para a geração de conteúdo dinâmico
 - São convertidas em servlets quando acessadas pela primeira vez
- JavaBeans
 - Componentes tradicionais em Java que podem ser usados em servlets e JSPs

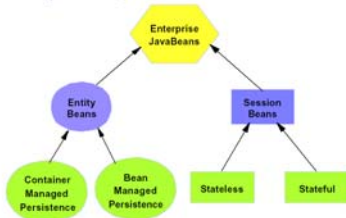
Java 2 Enterprise Edition (Camadas)

Componentes J2EE na Camada de Aplicação

- Componentes da camada de aplicação são chamados Enterprise Java Beans (EJB)
- Há vários tipos de EJBs:
- Session Beans
 - Representam uma conversa transiente com um cliente
 - Quando o cliente termina, a session bean some
- Entity Bean
 - Representam dados persistentes gravados num banco de dados (tipicamente uma linha de uma tabela)
- Message-Driven Bean
 - Uma combinação de um session bean com um Listener de mensagem Java Message Service (JMS)
 - Permite que um componente de aplicação (o message bean) receba mensagens assíncronas
 - Não falaremos desse tipo de Bean nesta disciplina

Java 2 Enterprise Edition (Camadas)

Taxonomy: Enterprise JavaBeans V1.1

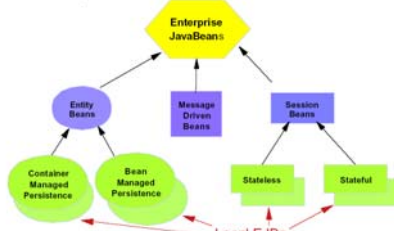


Sistemas Distribuídos 2007
Prof. Carlos Paes

10

Java 2 Enterprise Edition (Camadas)

Taxonomy: Enterprise JavaBeans V2.0



Sistemas Distribuídos 2007
Prof. Carlos Paes

11

Java 2 Enterprise Edition (Camadas)

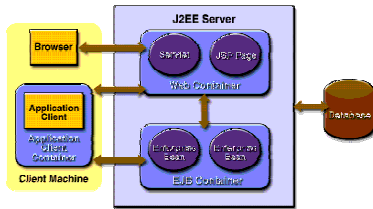
A camada de dados

- Observe que a camada de chamamos "de dados" pode ser um banco de dados ou outra coisa:
 - Por exemplo, pode ser um sistema ERP, CRM ou outro sistema legado
 - Por esse motivo, a camada frequentemente é chamada de "camada EIS"

Sistemas Distribuídos 2007
Prof. Carlos Paes

12

Arquitetura J2EE



J2EE Server and Containers

Arquitetura J2EE Containers e Serviços

- ✎ O programador se concentra no Business Logic, um middleware faz todo o trabalho automaticamente
- ✎ A entidade que faz essa mágica é o Container
- ✎ Um container "envolve" um componente de forma a capturar mensagens dirigidas ao componente e fornecer serviços automáticos a este

Arquitetura J2EE Containers e Serviços

- ✎ Portanto, antes de ser usado, um componente (seja cliente, Web ou EJB) deve:
 - Ser montado numa aplicação
 - Ser "deployed" (implantado) dentro de um container
- ✎ O container pode ser configurado em tempo de deployment
 - Com declarative programming, isto é, mudança de atributos

Arquitetura J2EE

Containers e Serviços

Exemplos do que se faz no deployment ao configurar um container:

- Estabelecer segurança
- Estabelecer o tratamento transacional
- Mapear nomes entre a aplicação e os recursos disponíveis

O container também gerencia serviços não configuráveis:

- O lifecycle dos componentes (achar, criar, destruir, ...)
- Pooling de recursos (conexões de bancos de dados, por exemplo)
- Persistência de dados

Arquitetura J2EE

Containers e Serviços

Tipos de Containers

- Os seguintes tipos de containers existem e executam no servidor J2EE:
- **Container EJB**: um tal container para acolher algumas ou todas as Enterprise Beans (EJBs) de uma aplicação
- **Web container**: um tal container para acolher algumas ou todas as JSPs e servlets de uma aplicação

Arquitetura J2EE

Containers e Serviços

- Os seguintes tipos de containers existem e executam na máquina cliente:
 - Application Client Container: para executar uma aplicação "console"
- Observe que servlets e JSPs podem executar sem um "J2EE server" completo
 - Podem executar num servidor Web com suporte especial sem ter suporte a EJB
 - Por exemplo: Apache Tomcat

APIs J2EE

- ✚ **Java 2 Platform, Standard Edition (J2SE™)**
 - O antigo JDK
- ✚ **Enterprise JavaBeans Technology**
- ✚ **JDBC API**
 - Para acessar Bancos de Dados
- ✚ **Java Servlet Technology**
- ✚ **JavaServer Pages (JSP) Technology**
- ✚ **Java Message Service (JMS)**
 - Para comunicação assíncrona distribuída, fracamente acoplada e confiável

APIs J2EE

- ✚ **Java Transaction API (JTA)**
 - Para a demarcação de transações
- ✚ **JavaMail™ Technology**
 - Para que aplicações possam enviar mail
- ✚ **Java API for XML Processing (JAXP)**
 - Para implementar B2B, relatório XML, etc.
- ✚ **J2EE Connector Architecture**
 - Para se conectar de forma simples a vários sistemas de informação corporativos (ERP, BD)
- ✚ **Java Authentication and Authorization Service (JAAS)**
 - Para prover serviços de autenticação e autorização
