

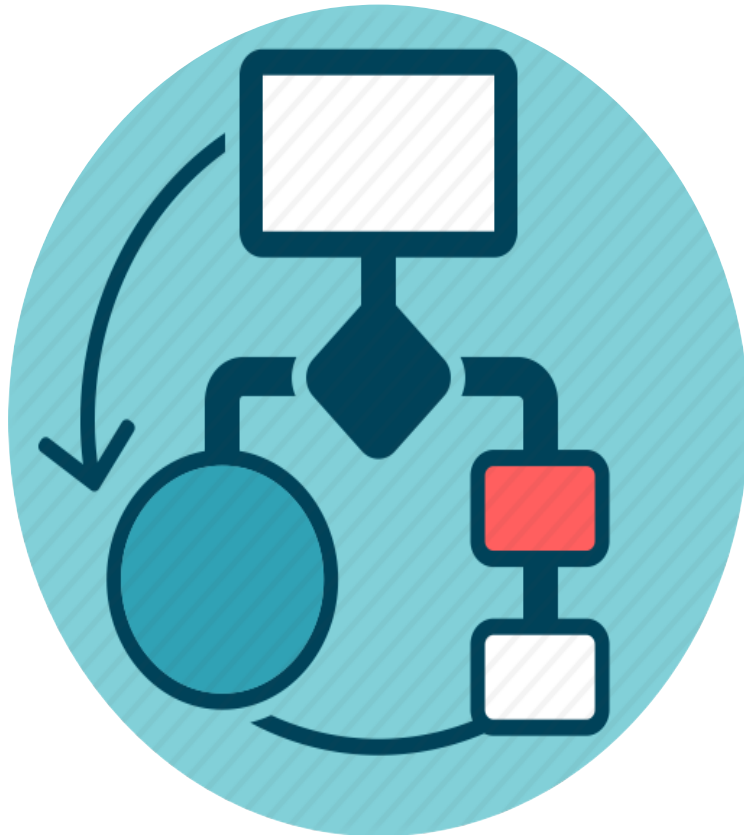


DIAGRAMA DE PERFIL

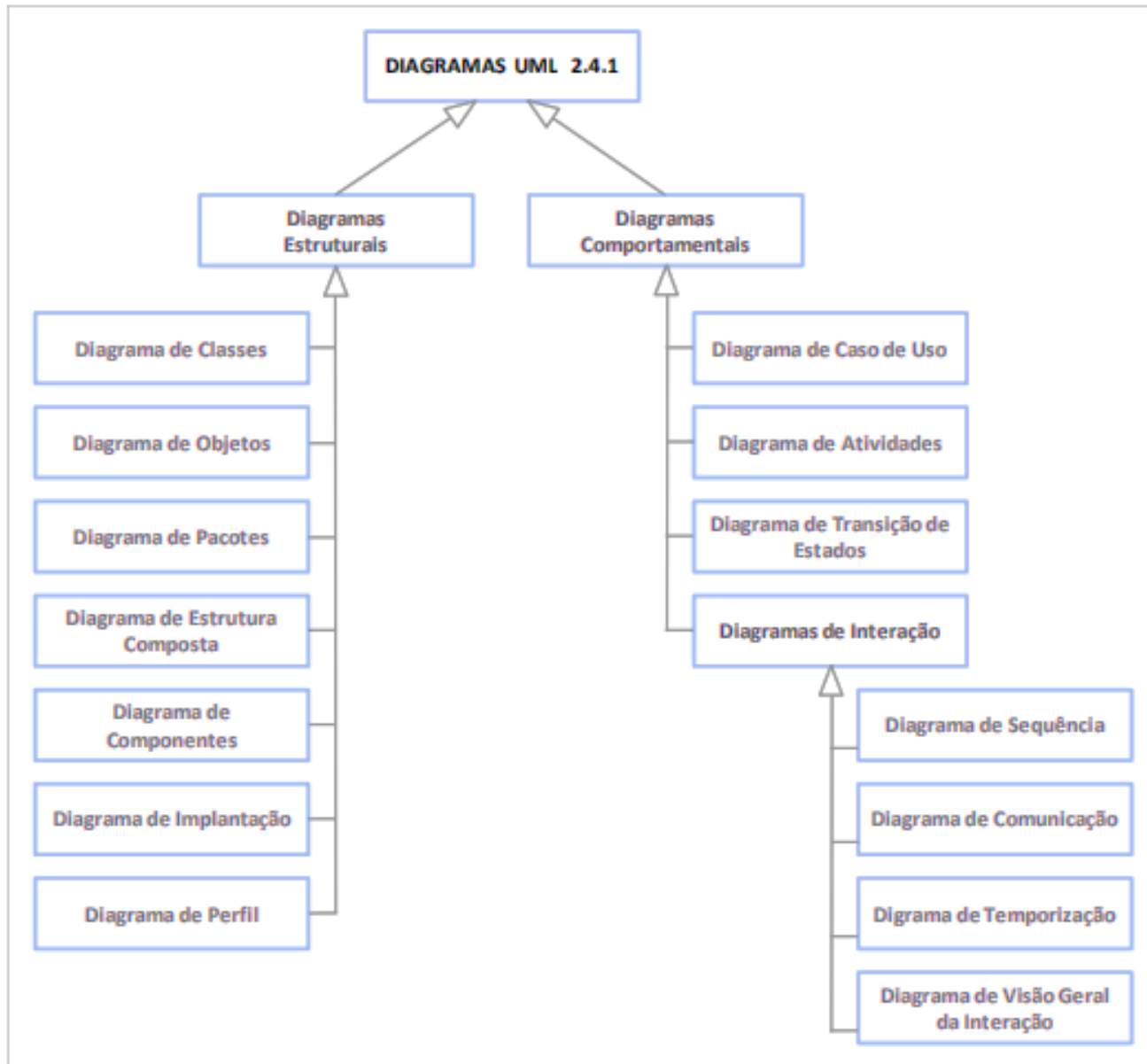
U

M

L

THAISSA DE OLIVEIRA

INTRODUÇÃO



O QUE É DIAGRAMA DE PERFIL?

- O diagrama de perfil é um diagrama UML auxiliar que permite definir estereótipos personalizados, valores marcados e restrições.
- Os diagramas de perfil foram introduzidos pela primeira vez na UML 2.0

O QUE É DIAGRAMA DE PERFIL?

- O mecanismo Profile foi definido na UML para fornecer um mecanismo de extensão leve ao padrão UML.
- Os perfis permitem adaptar o metamodelo UML para diferentes plataformas (como J2EE ou .NET) ou domínios (como modelagem de processos de negócios ou em tempo real).

O QUE É DIAGRAMA DE PERFIL?

- Por exemplo, a semântica dos elementos padrão do metamodelo UML pode ser especializada em um perfil.
- Em um modelo com o perfil "modelo Java", a generalização de classes deve poder se restringir à herança única sem ter que designar explicitamente um estereótipo «classe Java» para cada instância da classe.

FUNCIONAMENTO

O mecanismo de perfis não é um mecanismo de extensão de primeira classe. Ele não permite modificar os metamodelos existentes ou criar um novo metamodelo como o MOF. O perfil permite apenas a adaptação ou customização de um metamodelo existente com construções específicas para um domínio, plataforma ou método específico. Não é possível remover nenhuma das restrições que se aplicam a um metamodelo, mas é possível adicionar novas restrições específicas ao perfil.

FUNCIONAMENTO

- As personalizações de metamodelos são definidas em um perfil, que é aplicado a um pacote. Estereótipos são metaclasses específicas, valores marcados são metaatributos padrão e perfis são tipos específicos de pacotes.
- Os perfis podem ser aplicados dinamicamente ou retirados de um modelo. Eles também podem ser combinados dinamicamente para que vários perfis sejam aplicados ao mesmo tempo no mesmo modelo.

PRINCIPAIS ELEMENTOS

- Os nós e arestas gráficos usados nos diagramas de perfil são: perfil, metaclasse, estereótipo, extensão, referência, aplicação de perfil.

ESTEREÓTIPOS

- Os estereótipos permitem aumentar o vocabulário da UML. Você pode adicionar, criar novos elementos de modelo, derivados dos existentes, mas que possuem propriedades específicas que são adequadas ao seu domínio de problema.
- Os estereótipos são usados para introduzir novos blocos de construção que falam o idioma do seu domínio e parecem primitivos. Permite introduzir novos símbolos gráficos. Por exemplo: Ao modelar uma rede, pode ser necessário ter símbolos para <<router>>, <<switches>>, <<hub>> etc. Um estereótipo permite fazer com que essas coisas pareçam primitivas.

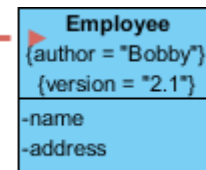
VALORES MARCADOS TAGGED VALUES

- Os valores com tags são usados para estender as propriedades da UML, para que você possa adicionar informações adicionais na especificação de um elemento do modelo. Permite especificar pares de valores de palavras-chave de um modelo em que as palavras-chave são os atributos. Os valores marcados são renderizados graficamente como uma cadeia de caracteres entre colchetes.
- Por exemplo: considere uma equipe de liberação responsável pela montagem, teste e implantação de um sistema. Nesse caso, é necessário acompanhar a versão e testar os resultados do subsistema principal. Valores marcados são usados para adicionar essas informações.

VALORES MARCADOS TAGGED VALUES

- Pode ser útil para adicionar propriedades ao modelo para alguns propósitos úteis:
 - Geração de código
 - Controle de versão
 - Gerenciamento de configurações
 - Autoria
 - Etc

Two tagged values — — —

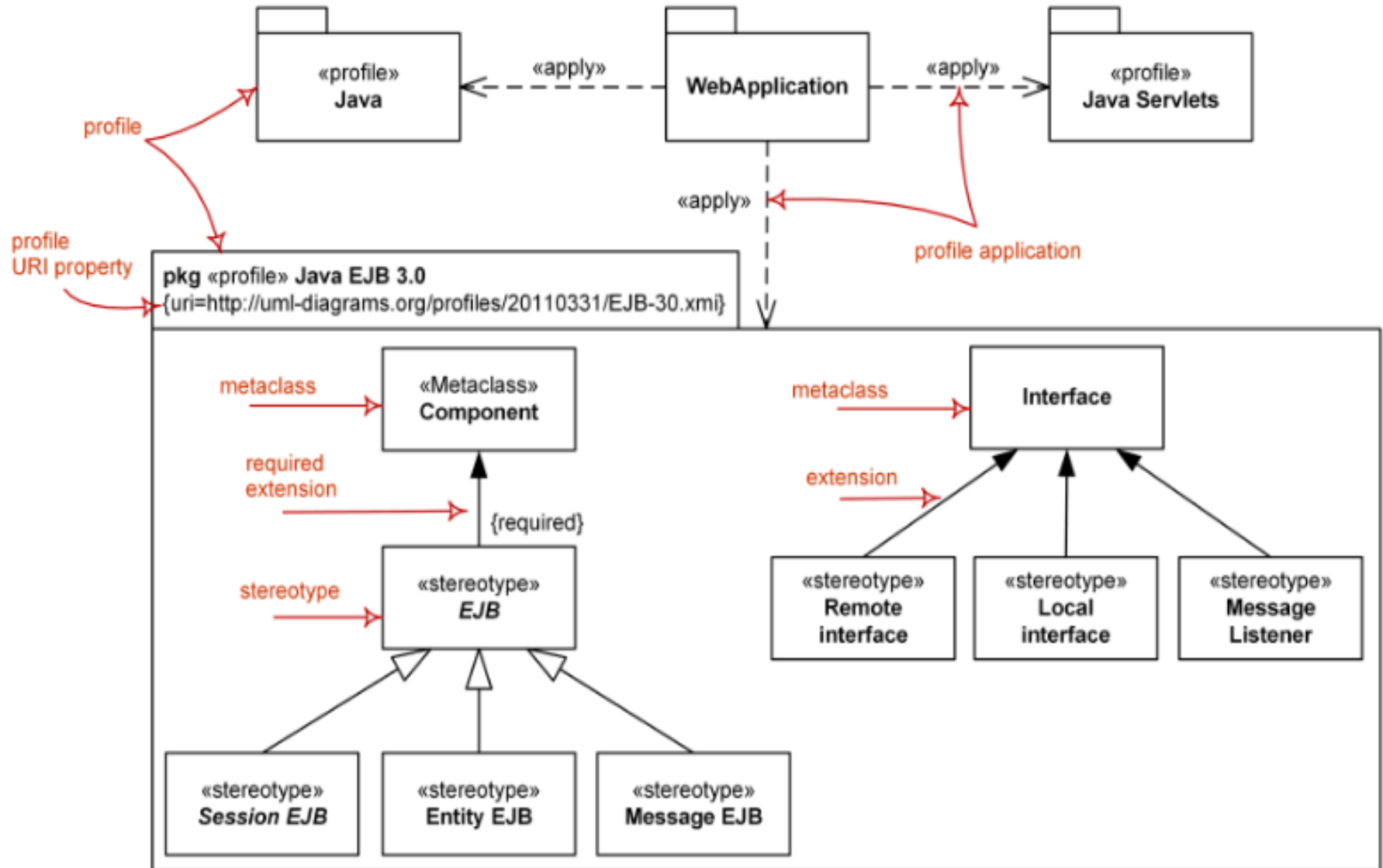


RESTRIÇÕES

- São as propriedades para especificar semânticas ou condições que devem ser mantidas verdadeiras o tempo todo. Ele permite que você estenda a semântica do bloco de construção UML adicionando novos protocolos. Graficamente, uma restrição é renderizada como uma cadeia de caracteres entre colchetes colocados perto do elemento associado.
- Por exemplo: No desenvolvimento de um sistema em tempo real, é necessário adornar o modelo com algumas informações necessárias, como tempo de resposta. Uma restrição define um relacionamento entre os elementos do modelo que devem ser usados {subconjunto} ou {xor}. As restrições podem estar em atributos, atributos derivados e associações. Também pode ser anexado a um ou mais elementos do modelo mostrados como uma nota.

PRINCIPAIS ELEMENTOS DO DIAGRAMA DE PERFIL

UML - PERFIL, ESTEREÓTIPO, METACLASSE, EXTENSÃO, APLICATIVO DE PERFIL.

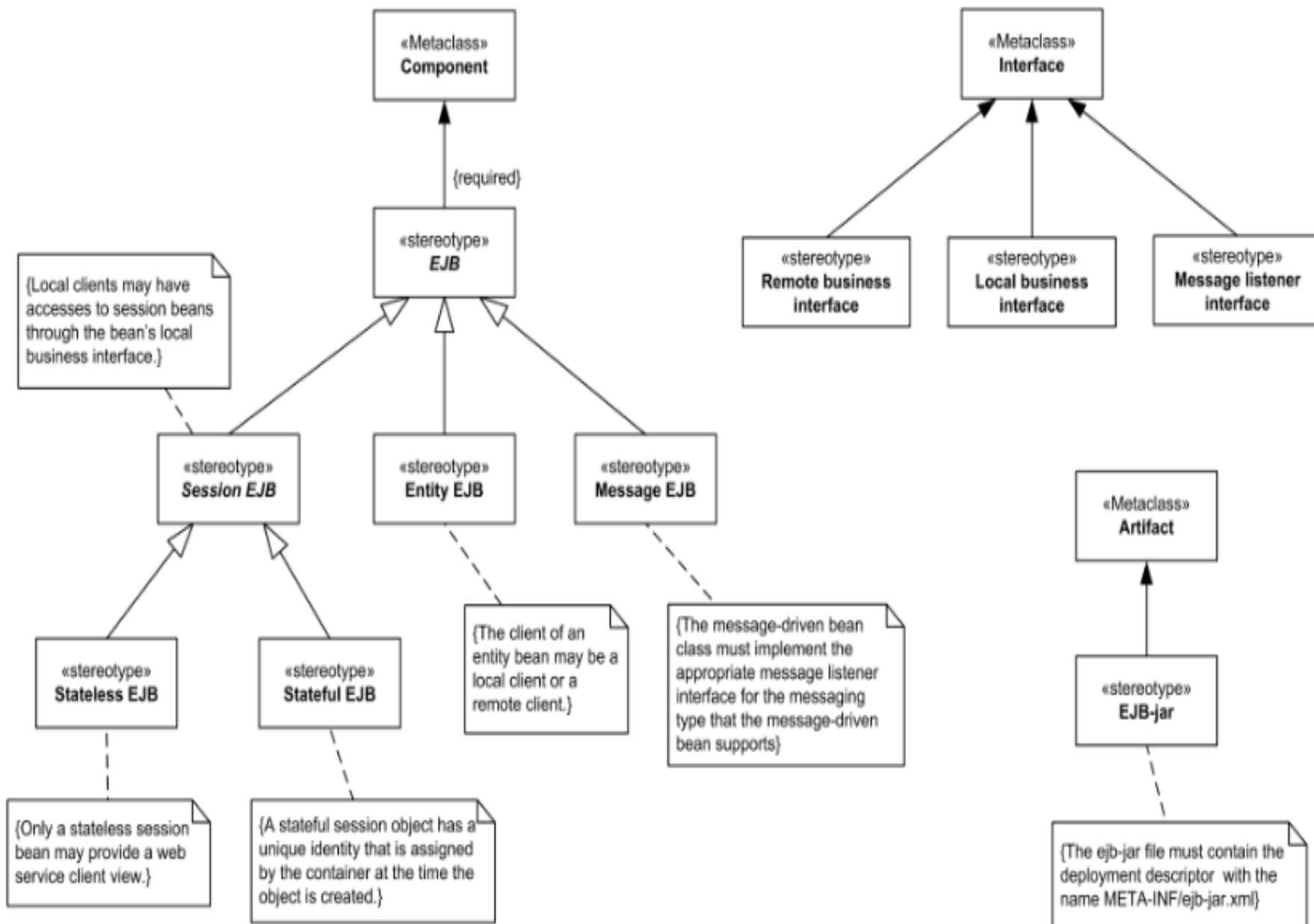


EXEMPLOS:

ENTERPRISE JAVABEANS™

EJB 3.0

- Um exemplo de perfil UML simplificado e não oficial para Enterprise JavaBeans™ (EJB) 3.0 com suporte para Enterprise JavaBeans de sessão, entidade e orientado a mensagens.
- Metamodelo oficial e perfil UML para Java e EJB Especificação, Versão 1.0 por OMG e JCP [Perfil UML para Java e EJB. A versão 1.0] é para Java 1.3, EJB 1.1 e, provavelmente, UML 1.4, portanto, pode ser de algum interesse para os bookworms.



EXEMPLOS:

ENTERPRISE JAVABEANS™

EJB 3.0

- A especificação UML 2.4 [UML 2.4 - Superestrutura] fornece um exemplo de perfil para a versão não especificada do J2EE / Enterprise Java Beans (EJB) no Anexo D.1. Esse perfil de amostra UML não é normativo (oficial) nem completo e é fornecido apenas como ilustração.
- A especificação EJB 3.0 define os beans de sessão com e sem estado. Existem diferenças na API entre beans de sessão com estado e beans de sessão sem estado. O cliente de um bean de sessão pode ser um cliente local, um cliente remoto ou um cliente de serviço da web, dependendo da interface fornecida pelo bean e usada pelo cliente.

EXEMPLOS:

ENTERPRISE JAVABEANS™

EJB 3.0

- O bean de sessão com estado representa uma sessão de conversação com um cliente específico. Esses objetos de sessão mantêm automaticamente seu estado conversacional em vários métodos invocados pelo cliente.
- Um objeto de entidade representa um objeto persistente de baixa granularidade. O cliente de um bean de entidade pode ser um cliente local ou o cliente pode ser um cliente remoto.

EXEMPLOS:

ENTERPRISE JAVABEANS™

EJB 3.0

- A classe do bean acionado por mensagens deve implementar a interface do ouvinte de mensagens apropriada para o tipo de sistema de mensagens que o bean acionado por mensagens suporta ou especificar a interface do ouvinte de mensagens usando a anotação de metadados MessageDriven ou o elemento descritor de implementação do tipo de sistema de mensagens.

EXEMPLOS:

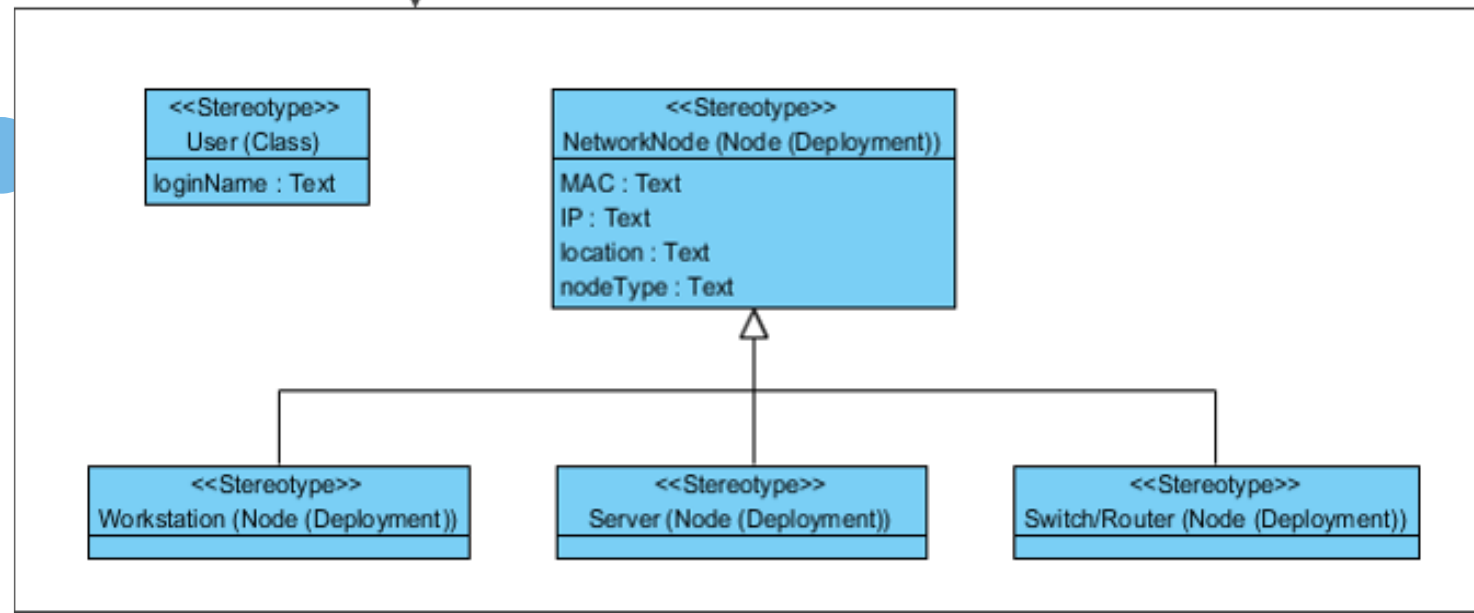
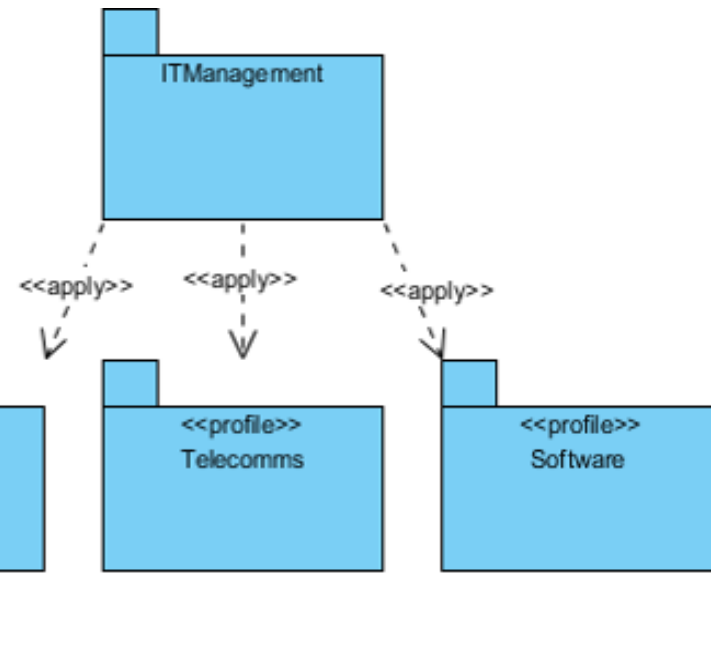
ENTERPRISE JAVABEANS™

EJB 3.0

- O arquivo ejb-jar deve conter o descritor de implementação no formato definido em Especificação EJB. O descritor de implementação deve ser armazenado com o nome META-INF / ejb-jar.xml. A classe do bean acionado por mensagens deve implementar a interface do ouvinte de mensagens apropriada para o tipo de sistema de mensagens que o bean acionado por mensagens suporta ou especificar a interface do ouvinte de mensagens usando a anotação de metadados MessageDriven ou o elemento descritor de implementação do tipo de sistema de mensagens.

EXEMPLOS: ENTERPRISE JAVABEANS™ EJB 3.0

- O arquivo ejb-jar deve conter o descritor de implementação no formato definido em Especificação EJB. O descritor de implementação deve ser armazenado com o nome META-INF / ejb-jar.xml.



OUTROS EXEMPLOS:

Service Oriented Architecture Modeling
Language (SoaML);

<https://www.uml-diagrams.org/soaml-uml-profile-diagram-example.html>

Digital Imaging and Communications
in Medicine (DICOM)

<https://www.uml-diagrams.org/dicom-uml-profile-diagram-example.html?context=prf-examples>

RELEMBRANDO

Em UML 2.3, o Diagrama de Perfil é um diagrama pertencente à categoria Diagrama de

- ☐ a) Estrutura estática, sendo usado para mostrar a estrutura de um sistema sob o nível mais baixo dos classificadores.
- ☐ b) Comportamento e mostra a estrutura interna de um classificador e o comportamento de uma colaboração.
- ☐ c) Comportamento, descrevendo a estrutura interna de uma classe e as colaborações que esta estrutura torna possível.
- ☐ d) Comportamento, sendo utilizado para descrever o *hardware* utilizado em implementações de sistemas e os ambientes de execução.
- ☐ e) Estruturas, que opera no nível metamodelo permitindo definir estereótipos personalizados, valores etiquetados e restrições.

ARTIGO

- UP4EG: Um Perfil UML para Modelagem de Jogos Educacionais Digitais
- Link:

<https://br-ie.org/pub/index.php/sbie/article/view/6692>



REFERÊNCIAS

- <https://www.uml-diagrams.org/profile-diagrams.html>
- <https://www.uml-diagrams.org/uml-22-diagrams.html>
- <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-profile-diagram/>
- <https://www.uml-diagrams.org/soaml-uml-profile-diagram-example.html>
- <https://www.uml-diagrams.org/java-ejb-uml-profile-diagram-example.html?context=prf-examples>
- <https://www.uml-diagrams.org/dicom-uml-profile-diagram-example.html?context=prf-examples>

