

Modelagem da Arquitetura

Diagramas de Implantação

Diagrama de Implantação

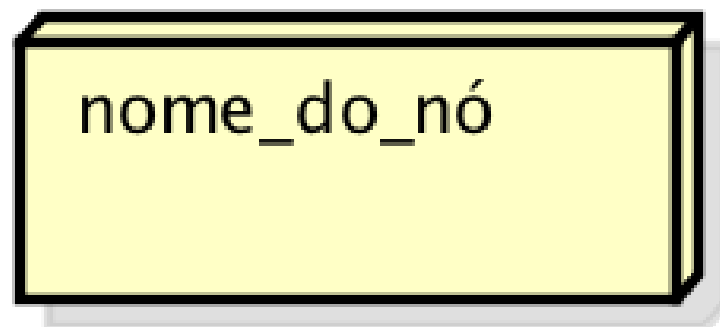
- Representa como é realizada a distribuição do sistema através de **nós de hardware**, componentes e dependências de software e as suas devidas relações de comunicação
- Modela o inter-relacionamento entre recursos de infraestrutura, de rede ou artefatos de sistemas (informação gerada pelo software)
- Os “nós” representam servidores físicos

Diagrama de Implantação

- Um nó é um elemento físico que existe em tempo de execução, e representam um recurso computacional com algum processamento
- Os nós são empregados na modelagem da “topologia” do hardware em que o sistema é executado

Representação Gráfica de um Nó

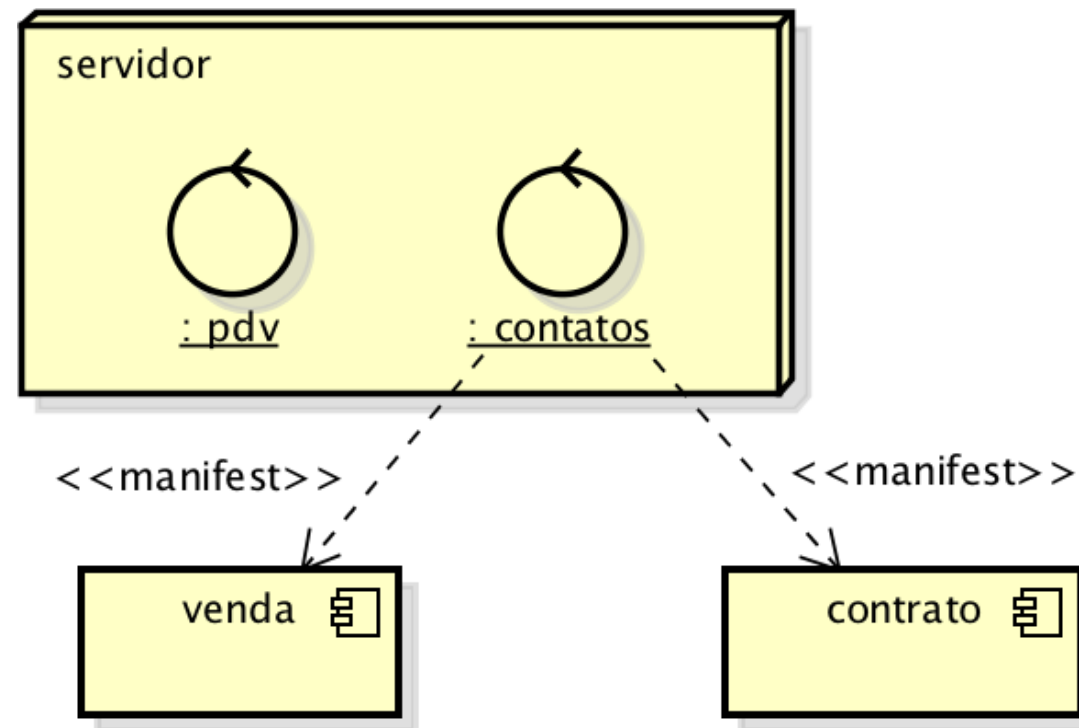
- A UML define a seguinte representação:



- Utilizando estereótipos (UML), podem ser especificados tipos específicos de processadores, dispositivos e plataformas de software

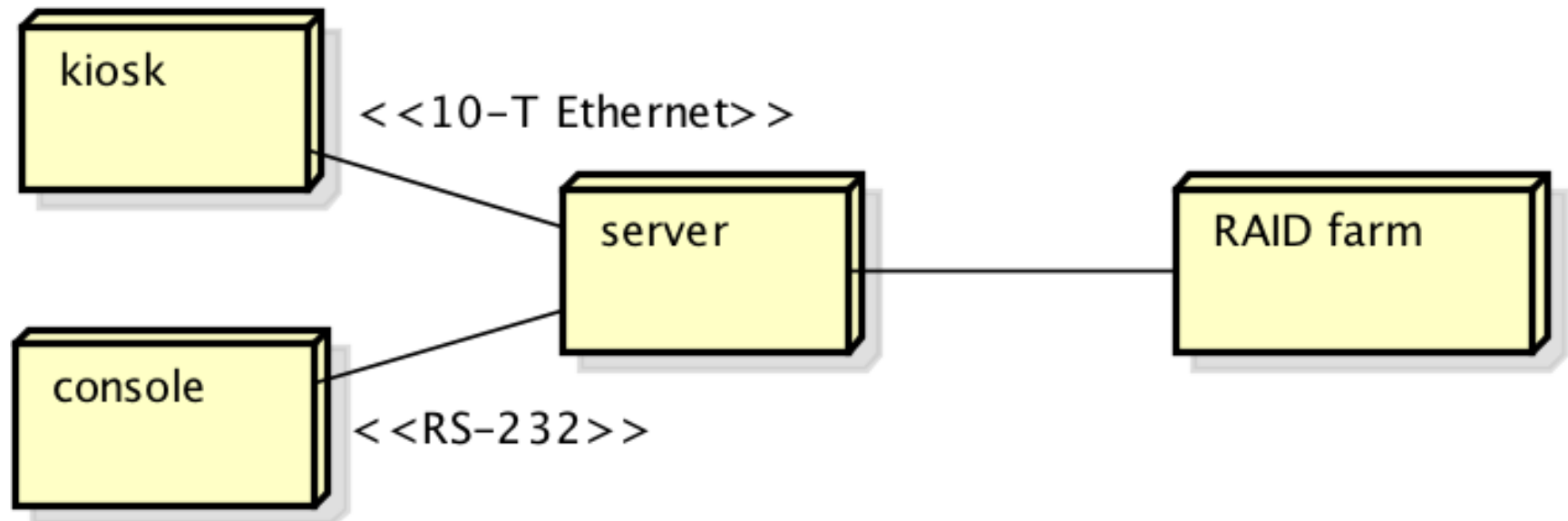
Nós e Artefatos

- Os artefatos são “itens” que participam da execução de um sistema – os nós são “itens” que executam os artefatos



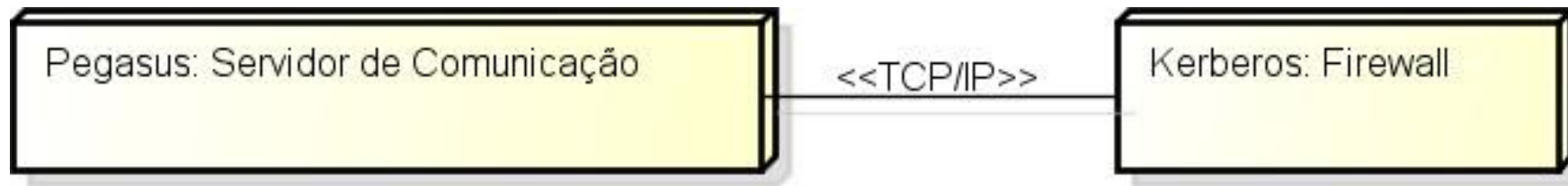
Conexões

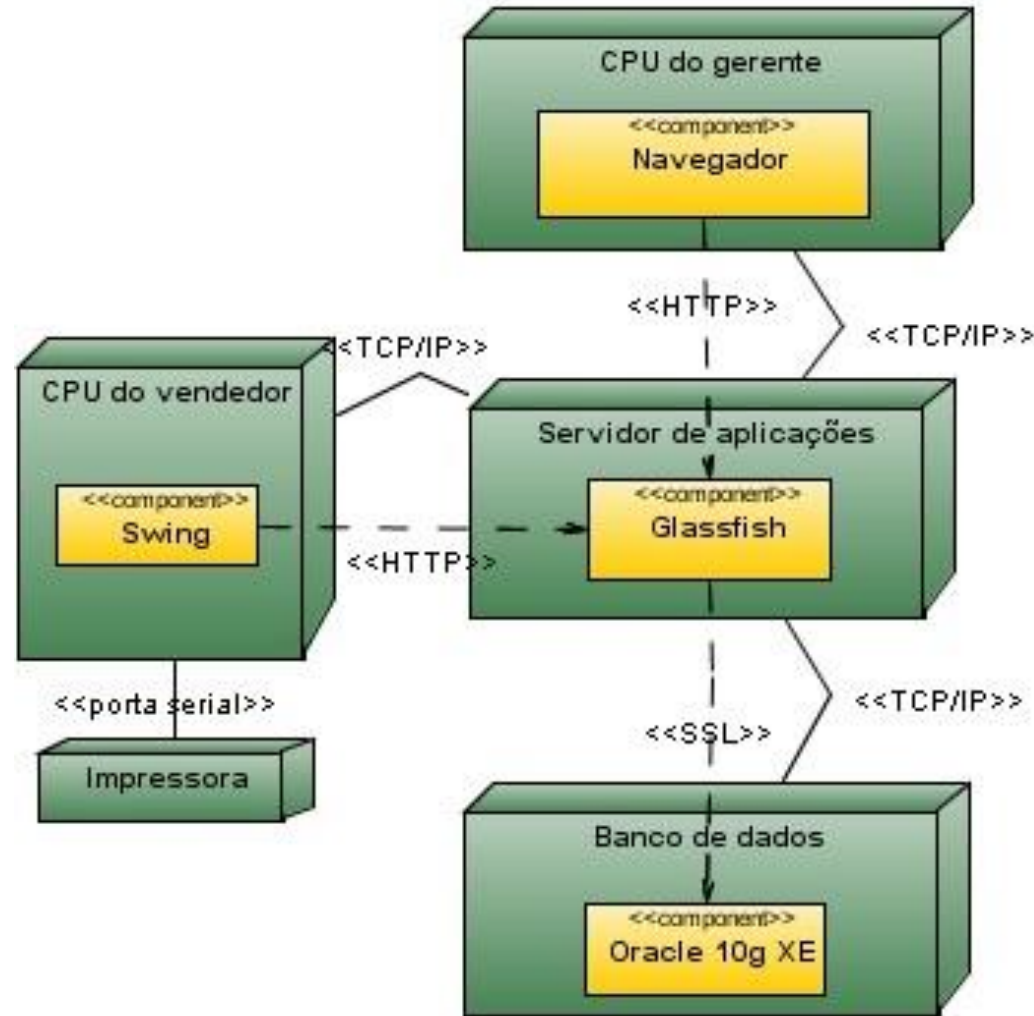
- Podem ser ilustrados os “relacionamentos” entre os nós, na forma de uma associação
 - Representa uma conexão física entre os nós
 - Os tipos de conexão são especificados por estereótipos

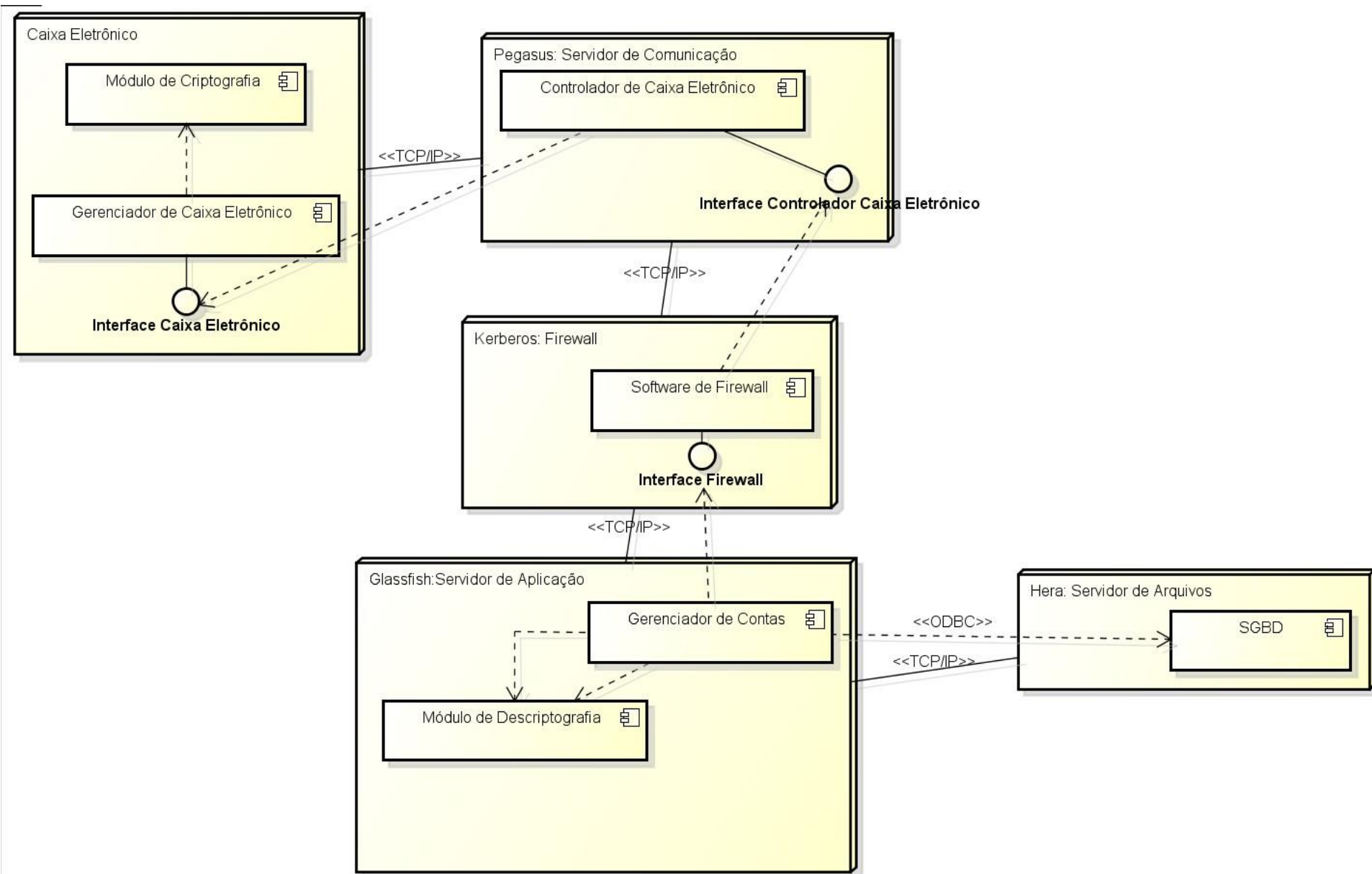


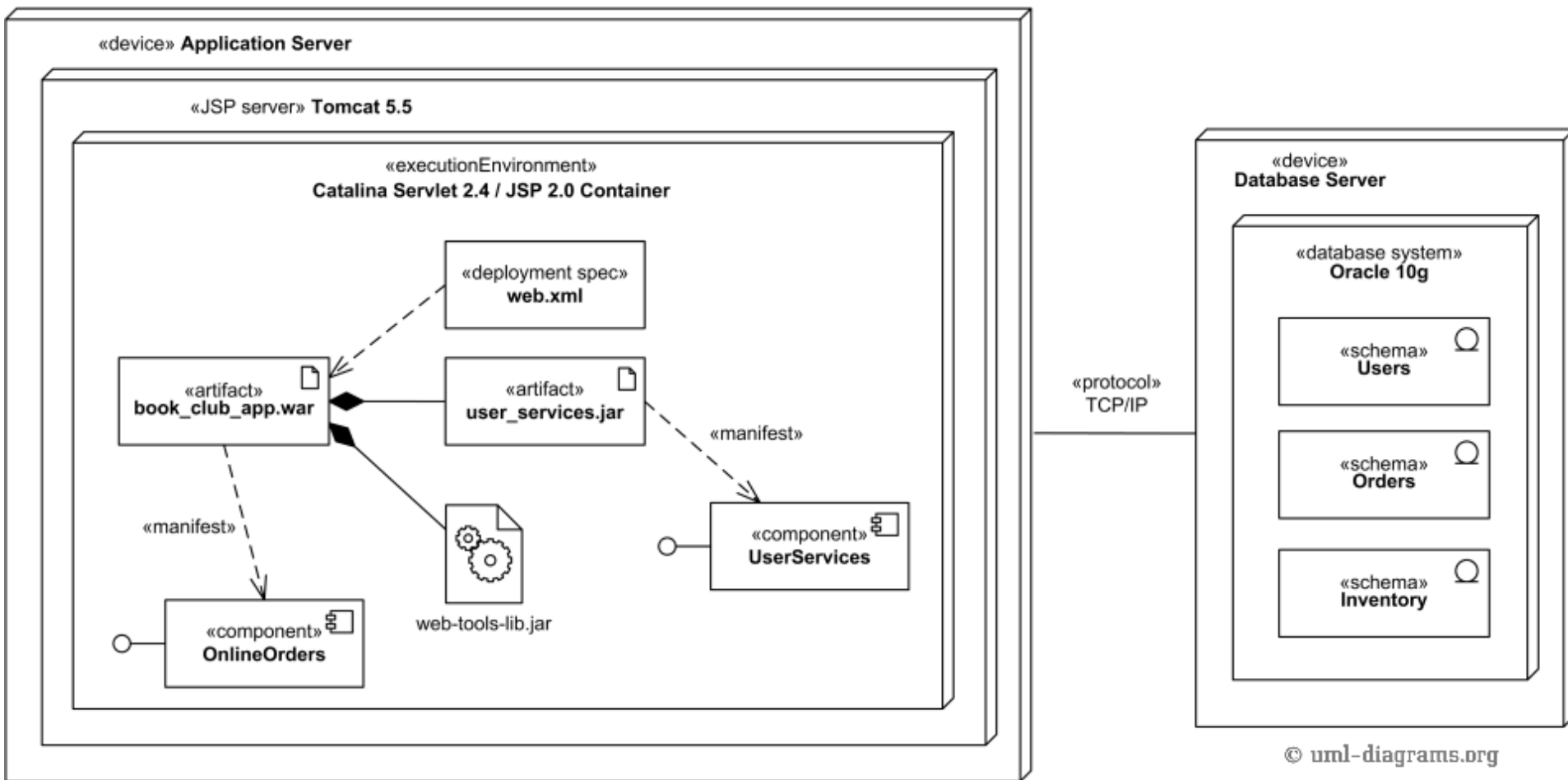
Conexões

- As associações indicam uma eventual “troca de informações”







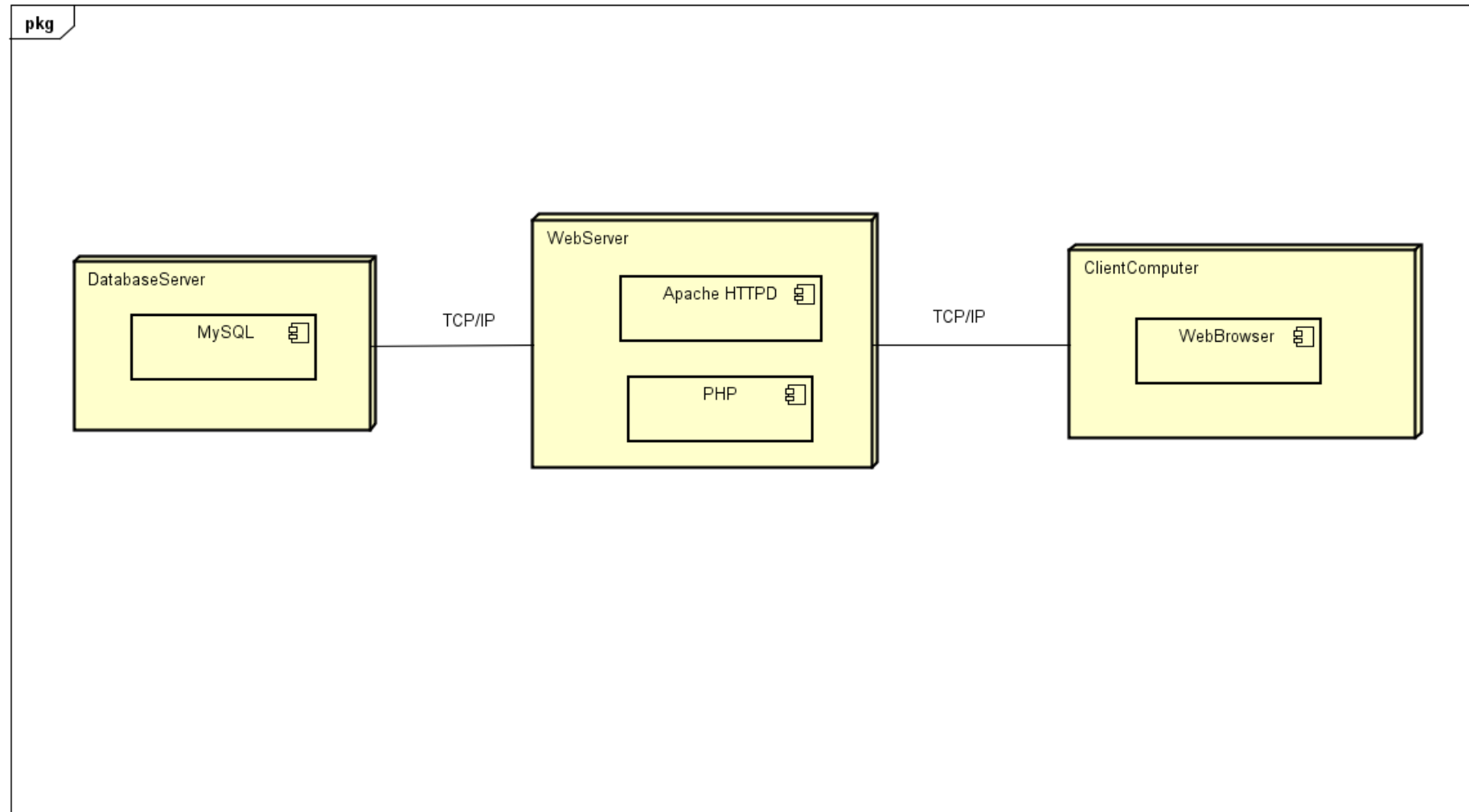


Exercício

Construa um Diagrama de Implantação para o seguinte cenário:

É preciso demonstrar que um sistema será composto por um servidor de banco de dados com uma base de dados utilizando o SGBD MySQL, um servidor de aplicação contendo o servidor Apache com o PHP instalado e o computador do cliente com um navegador. Para isso você precisará criar 3 nós (com os devidos componentes em cada um) e realizar a ligação entre eles.

Exercício



Modelagem da Arquitetura

Diagrama de Classes

Diagrama de Classes

- Diagramas de classes estão entre os tipos mais úteis de diagramas UML pois mapeiam de forma clara a estrutura de um determinado sistema ao modelar suas classes, seus atributos, operações e relações entre objetos
- Bastante usado por engenheiros de software para documentar arquiteturas de software, os diagramas de classes são um tipo de diagrama da estrutura porque descrevem o que deve estar presente no sistema a ser modelado.

Diagrama de Classes

- A UML foi criada como um modelo padronizado para descrever uma abordagem de programação orientada ao objeto.
- Como as classes são os componentes básicos dos objetos, diagramas de classes são os componentes básicos da UML.
- Os diversos componentes em um diagrama de classes podem representar as classes que serão realmente programadas, os principais objetos ou as interações entre classes e objetos.

Diagrama de Classes

- A forma de classe em si consiste em um retângulo com três linhas. A linha superior contém o nome da classe, a linha do meio, os atributos da classe e a linha inferior expressa os métodos ou operações que a classe pode utilizar.
- Classes e subclasses são agrupadas juntas para mostrar a relação estática entre cada objeto.

Diagrama de Classes

Benefícios do diagrama de classes:

- Ilustrar modelos de dados para sistemas de informação, não importa quão simples ou complexo.
- Entender melhor a visão geral dos esquemas de uma aplicação.
- Expressar visualmente as necessidades específicas de um sistema e divulgar essas informações por toda a empresa.

Diagrama de Classes

Componentes básicos de um diagrama de classes:

- **Parte superior:** contém o nome da classe. Esta parte é sempre necessária, seja falando do classificador ou de um objeto.
- **Parte do meio:** contém os atributos da classe. Use esta parte para descrever as qualidades da classe. É necessário somente quando se descreve uma instância específica de uma classe.
- **Parte inferior:** inclui as operações da classe (métodos). Exibido em formato de lista, cada operação ocupa sua própria linha. As operações descrevem como uma classe interage com dados.

Diagrama de Classes

Modificadores de acesso de membro:

- **Public (+)** – Acessível a todas as classes e em todos os pacotes.
- **Private (-)** – Acessível somente na classe em que é criado.
- **Protected (#)** – Acessível às classes do mesmo pacote e através da herança (mesmo fora do pacote).
- **Default (~)** – Padrão, quando nenhum for definido. Acessível às classes do mesmo pacote. Não é herdado para as classes filhas fora do pacote.
- **Estatic (sublinhado)** – Atributo da classe e não do objeto. Há somente um para todos os objetos.

Diagrama de Classes

Interações: Hereditariedade (Herança)

- Também conhecida como generalização, é o processo de um secundário, ou subclasse, assumindo a funcionalidade de um primário, ou superclasse.
- É simbolizada por uma linha conectada reta com uma ponta de seta fechada apontando para a superclasse.

Diagrama de Classes

Interações: Hereditariedade (Herança)

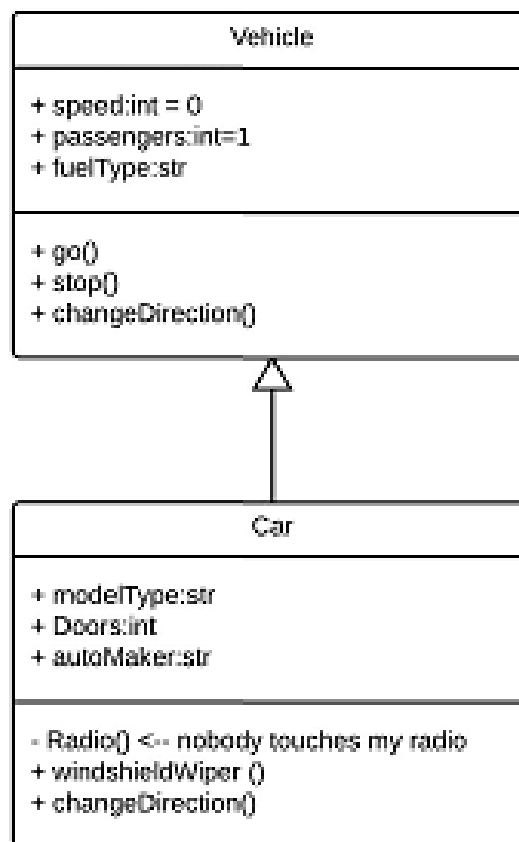


Diagrama de Classes

Interações: Associação bidirecional

- A relação padrão entre duas classes.
- Ambas as classes estão cientes uma da outra e de sua relação entre si.
- Essa associação é representada por uma linha reta entre duas classes.

Diagrama de Classes

Interações: Associação bidirecional

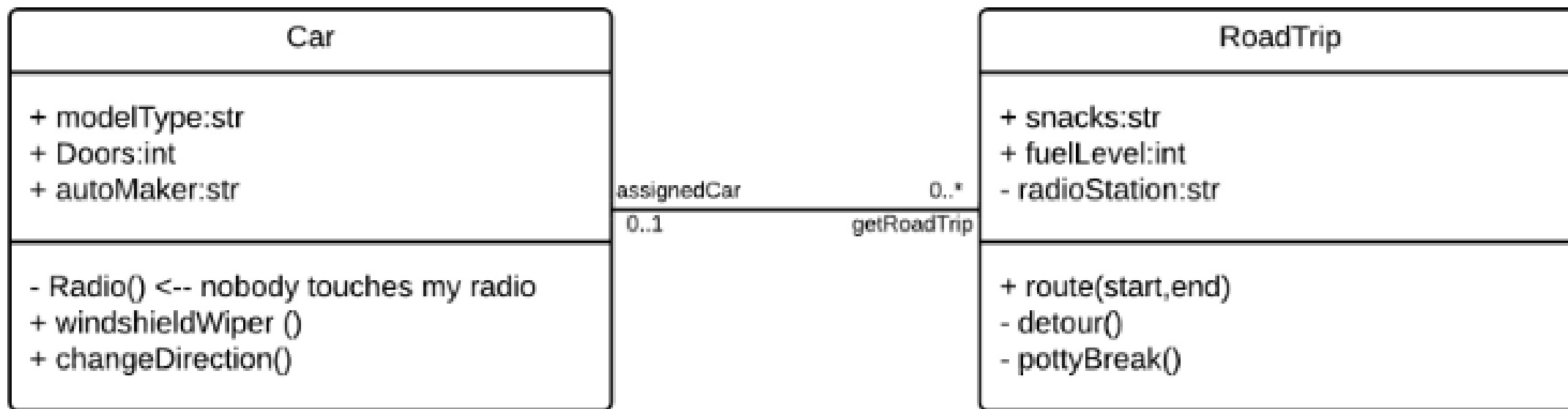


Diagrama de Classes

Interações: Associação unidirecional

- Uma relação um pouco menos comum entre duas classes.
- Uma classe tem conhecimento da outra e interage com ela.
- A associação unidirecional é modelada por uma linha reta de ligação com uma ponta de seta aberta da classe conhecimento à classe conhecida.

Diagrama de Classes

Interações: Associação unidirecional

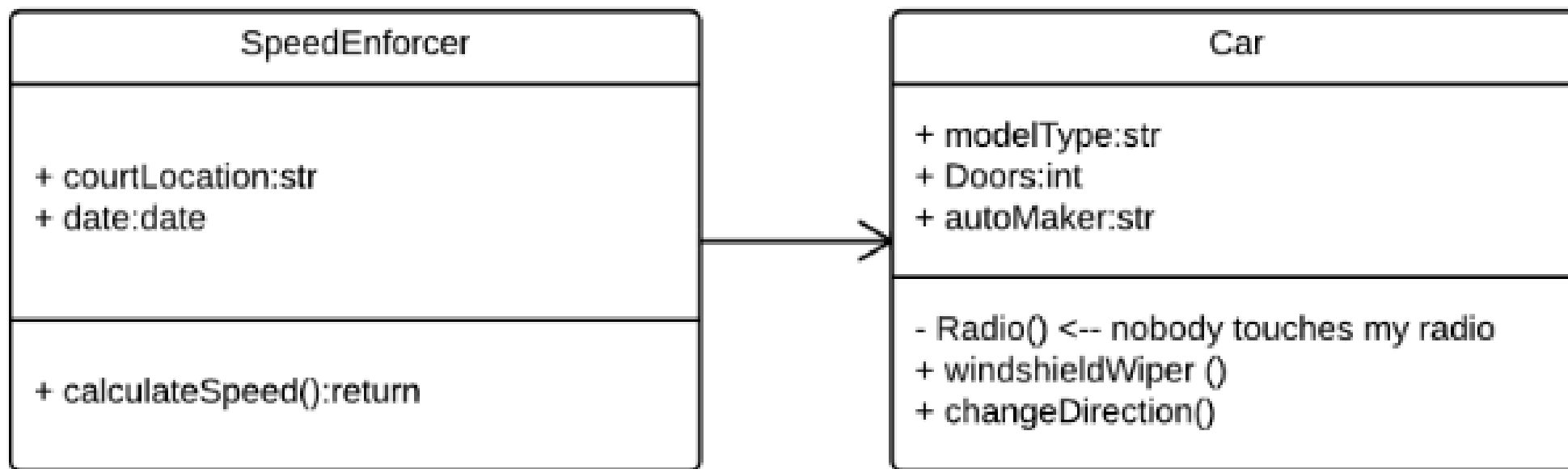


Diagrama de Classes

Interações: Agregação

- É um tipo especial de associação onde tenta-se demonstrar que as informações de um objeto (chamado objeto-todo) precisam ser complementados pelas informações contidas em um ou mais objetos de outra classe (chamados objetos-parte);
- O objeto-pai poderá usar as informações do objeto agregado

Diagrama de Classes

Interações: Agregação

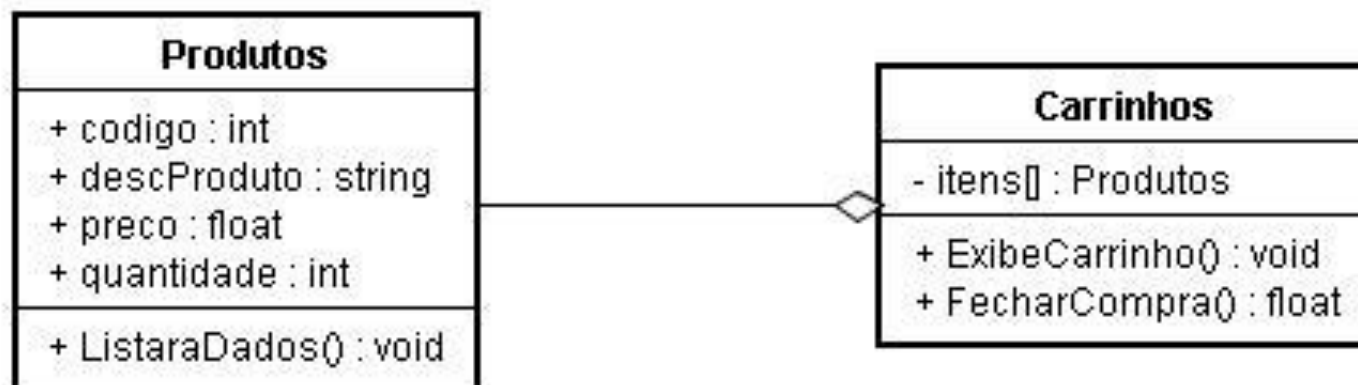


Diagrama de Classes

Interações: Composição

- A composição é uma variação da agregação.
- Uma composição tenta representar também uma relação todo-parte. No entanto, na composição o objeto-pai (todo) é responsável por criar e destruir suas partes. Em uma composição um mesmo objeto-parte não pode se associar a mais de um objeto-pai.

Diagrama de Classes

Interações: Composição

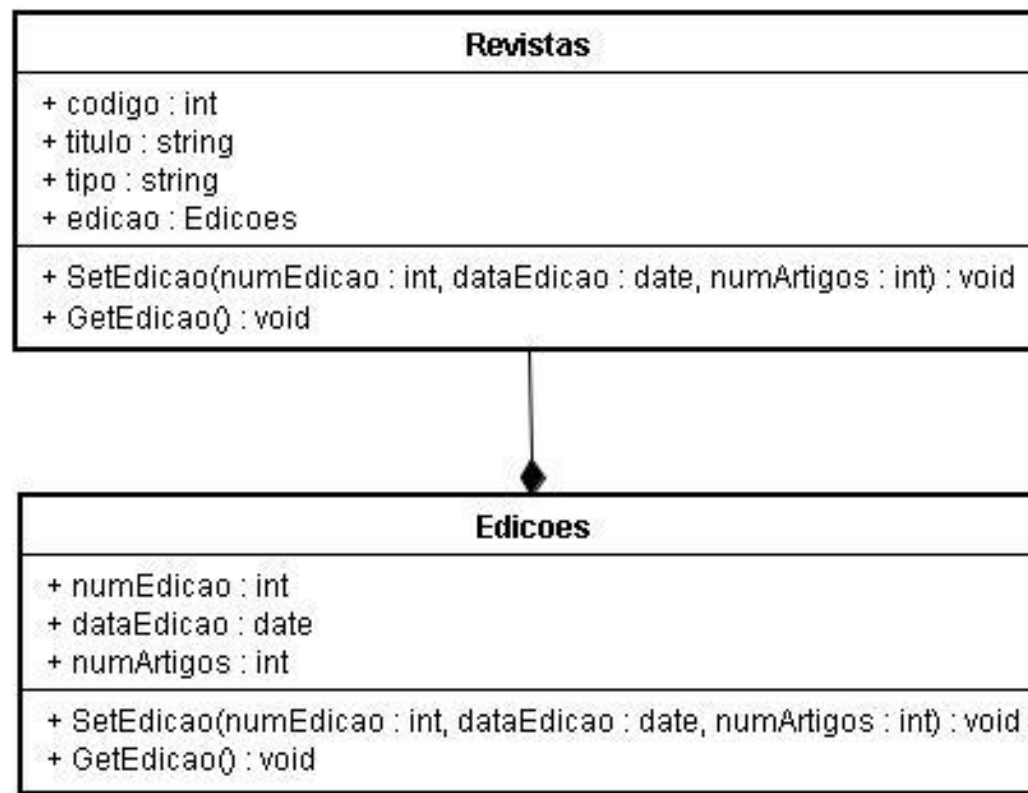


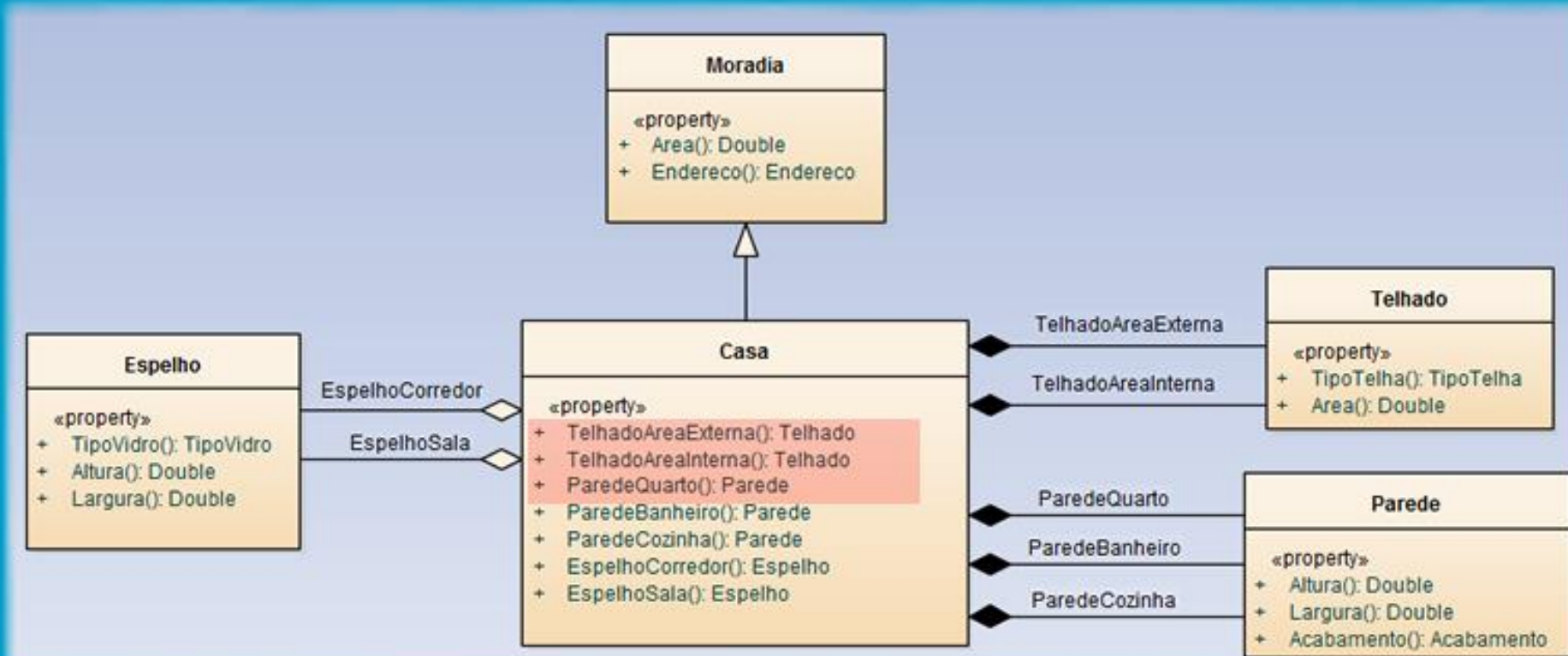
Diagrama de Classes

Herança x Agregação x Composição

- Vamos imaginar uma casa. Uma casa possui telhado e paredes, e não existe sem isso. Logo, uma casa é **composta** de telhado e paredes.
- Mas uma casa existe sem um espelho. Mas casas tem espelhos. Neste contexto, o Espelho é uma **Agregação**, agrega-se à Casa.
- Se a casa dependesse do Espelho para existir, então seria uma **Composição**.
- E se a casa fosse feito de Espelhos, fosse esta sua matéria prima, então seria uma **Herança**.

Diagrama de Classes

Herança x Agregação x Composição



Fonte dos Slides

- <https://www.lucidchart.com/pages/pt/o-que-e-diagrama-de-classe-uml>
- http://www.cpscetec.com.br/adistancia/poo_php/aula5.html