

Análise e Design Orientado a Objetos utilizando a UML (*Unified Modeling Language*)

Modelagem utilizando a UML

Professor *Sergio Akio Tanaka*
e-mail sergio.tanaka@unifil.br

Diagramas de Componentes

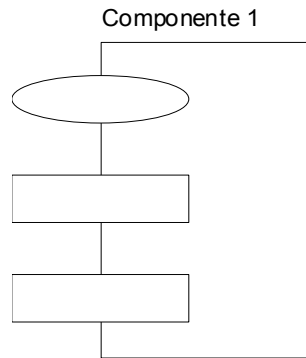
Diagramas de Componentes

- ▶ Um componente é uma parte física e substituível de um sistema ao qual se adapta e fornece a realização de um conjunto de interfaces. Graficamente, um componente é representado como um retângulo com abas.

Diagramas de Componentes

- ▶ Cada componente é representado por um ícone;
- ▶ Os componentes devem ser nomeados;
- ▶ E a dependência entre os componentes são indicados por notação de fluxo.
- ▶ Um diagrama de Pacotes pode representar um componente ou um *framework*

◆ Notação:



Entendendo o Conceito de Interface de Implementação

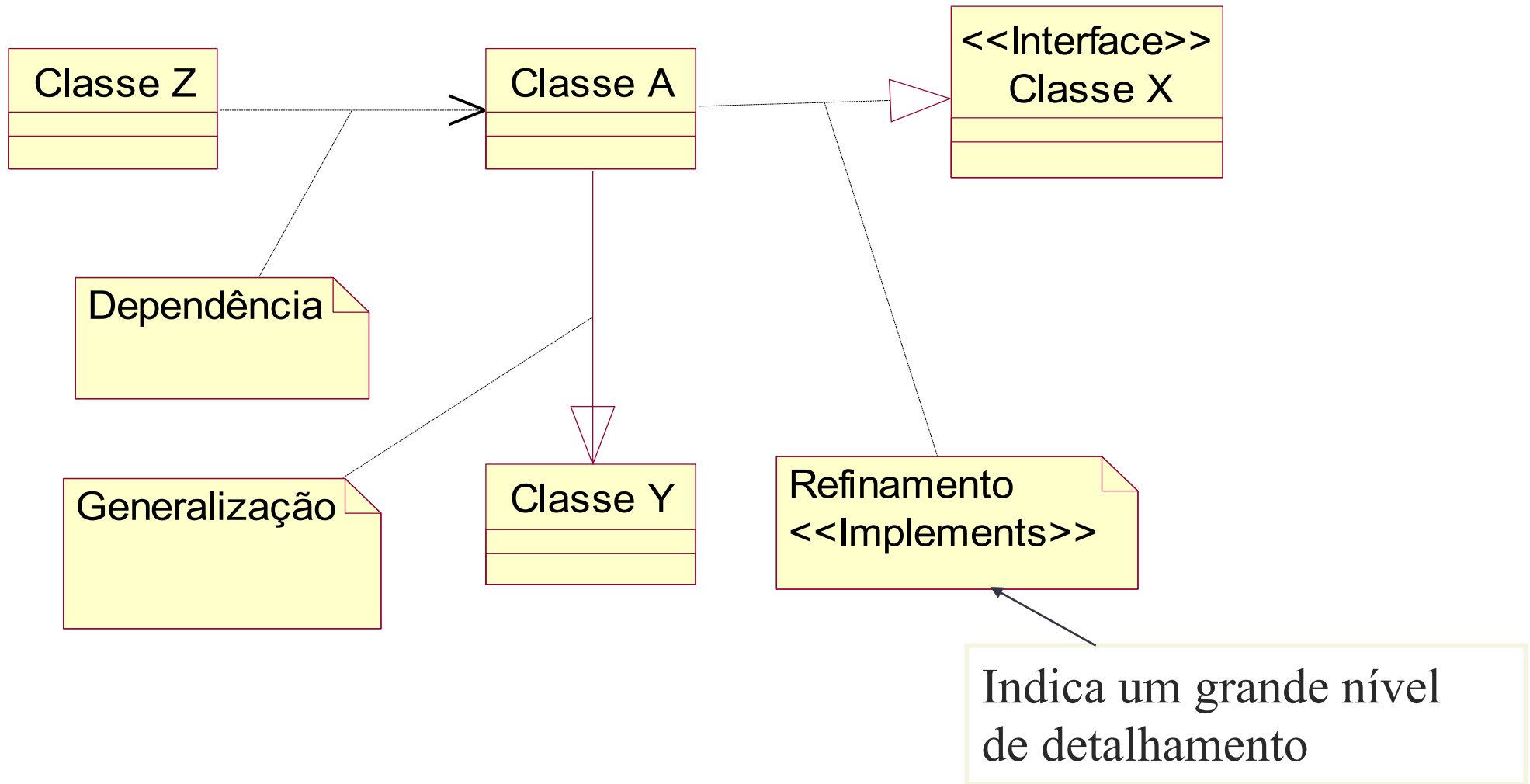
Interfaces na UML 2.0

- As interfaces são definidas no pacote
Superstructure::Classes::Interfaces

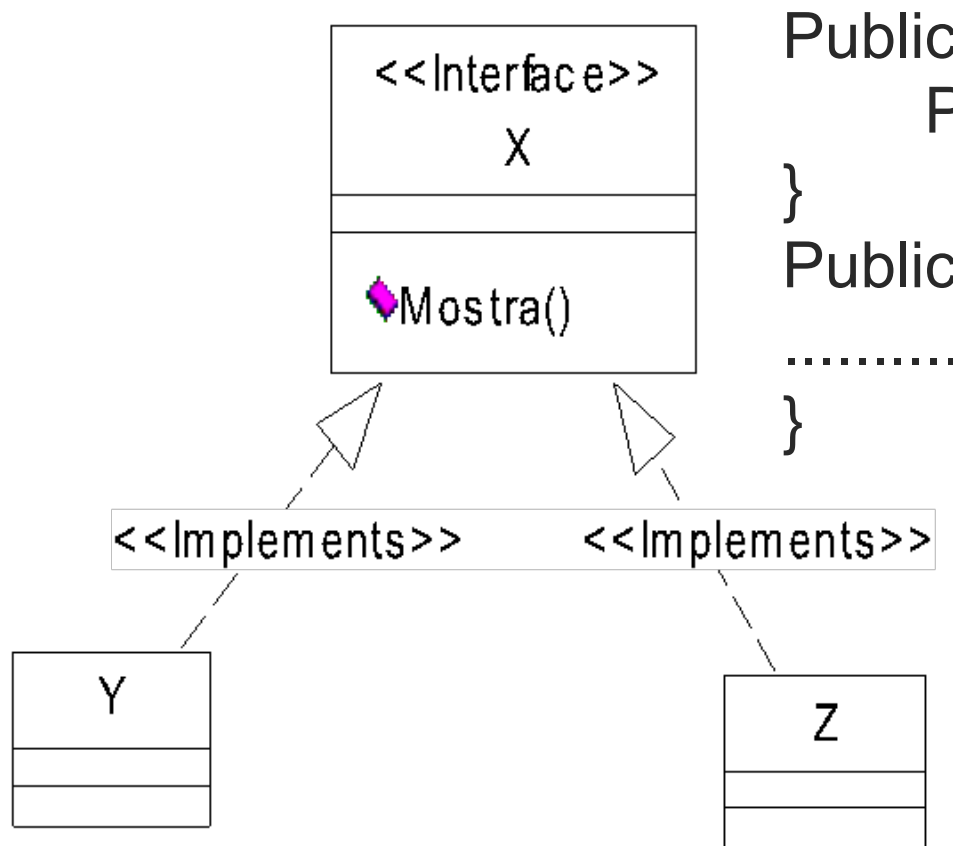
Interface de Implementação

- ▶ Uma interface é um especificador para operações externamente visíveis de uma classe ou um componente sem especificação de sua estrutura interna;
- ▶ Cada interface especifica apenas uma parte do comportamento de uma classe e não tem implementação;
- ▶ Equivalente a uma classe abstrata sem atributos, estados ou associações, porém possuem métodos abstratos;
- ▶ Pode-se utilizar o estereótipo <<interface>>.

Interface



Interface



```
Public Interface X{
    Public void mostra();
}
```

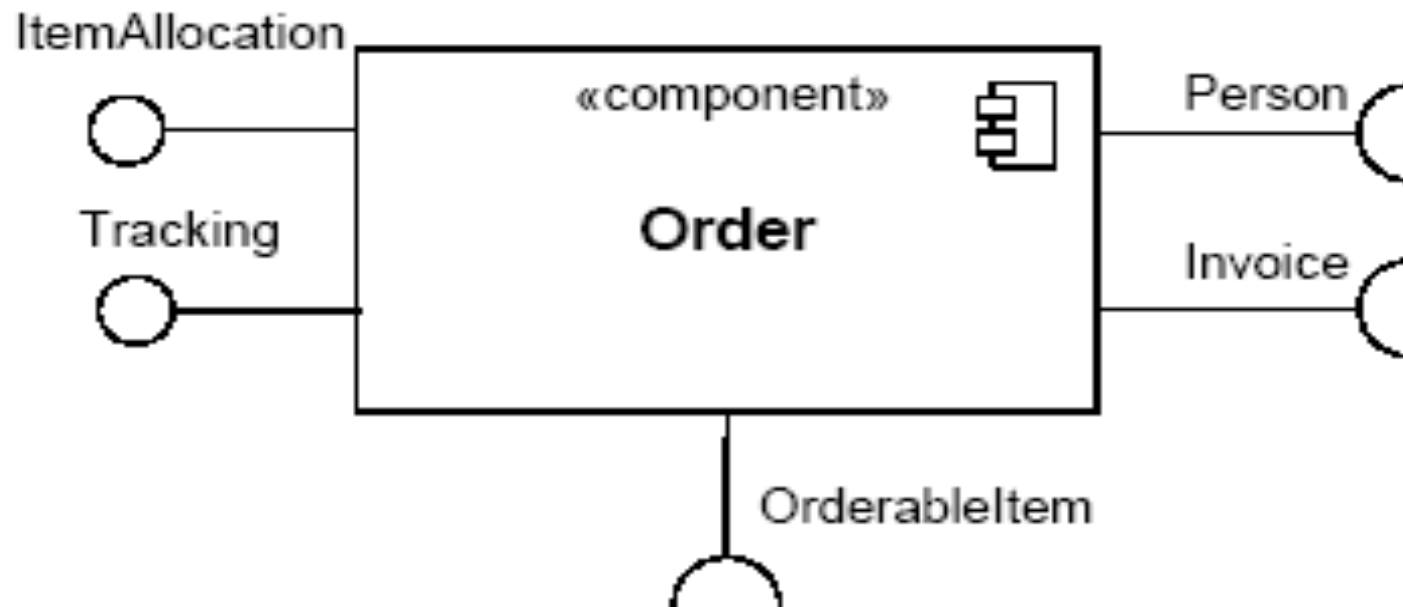
```
Public class Y extends W implements X{
    .....
}
```

Tipos de Interfaces

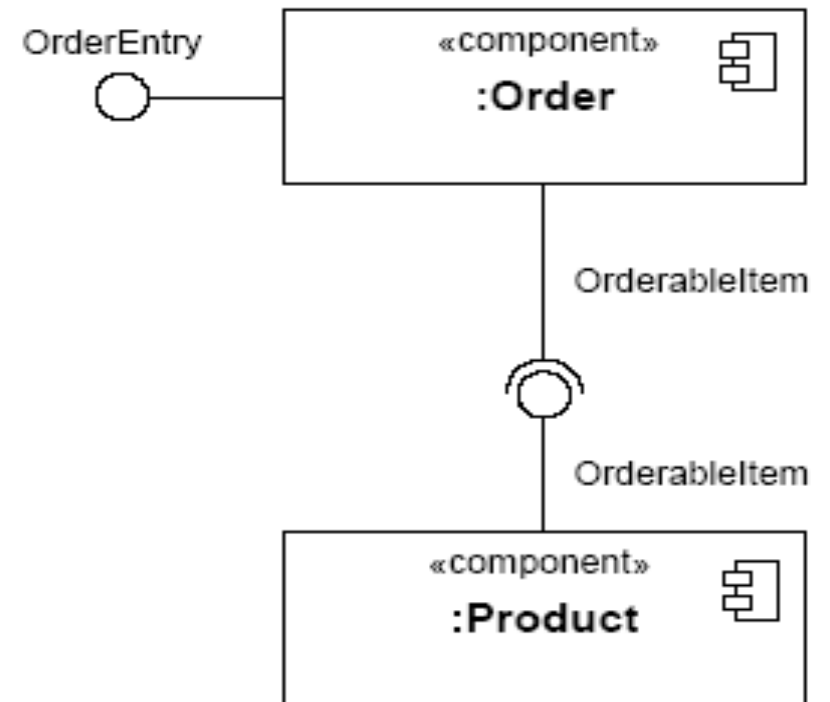
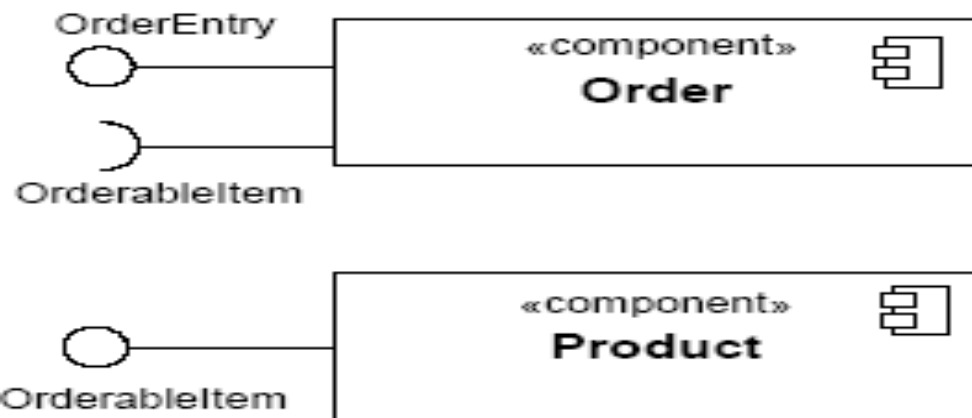
- Fornecidas: é um serviço que um objeto coloca a disposição de outros objetos. Um componente precisa pedir acesso a um serviço fornecido.
- Exigidas: quando um objeto exige acesso a um serviço. Um componente pode precisar de ajuda de outros componentes, e precisa definir exatamente o que ele precisa, e por isso define uma interface indicando o tipo de ajuda Exigida (modelada como uma meia circunferência)

INTERFACE UML 2.0

- Interface Exigida (Person, Invoice e OrderableItem)
- Interface Fornecida (itemAllocation, Tracking)



Conexão entre componentes



Diagramas de Componentes UML 1.x

► Notação de Dependência:

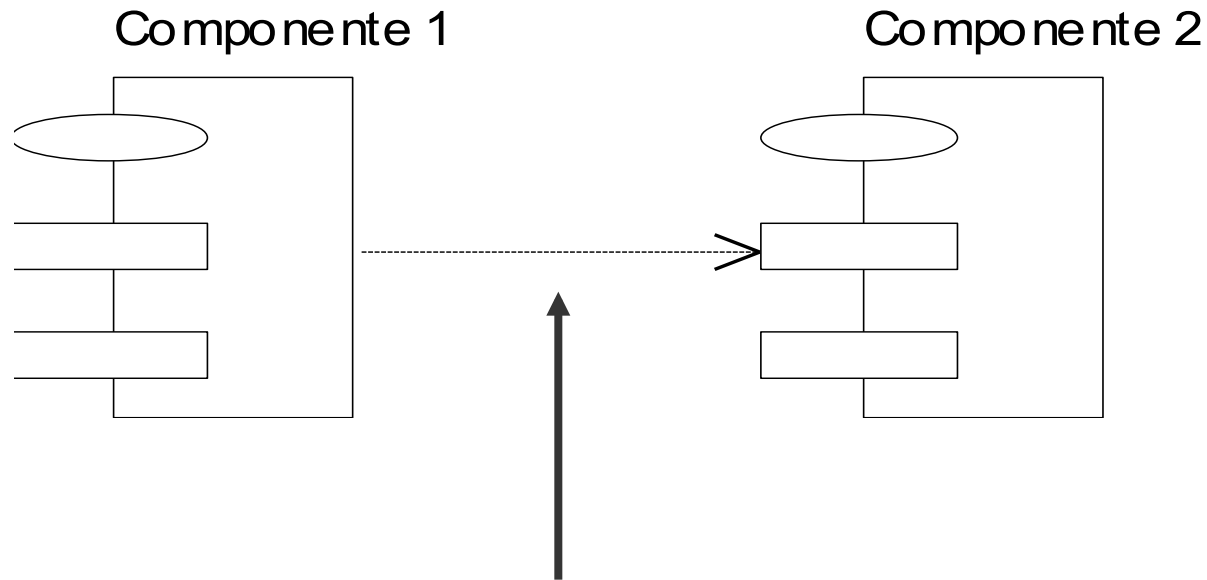
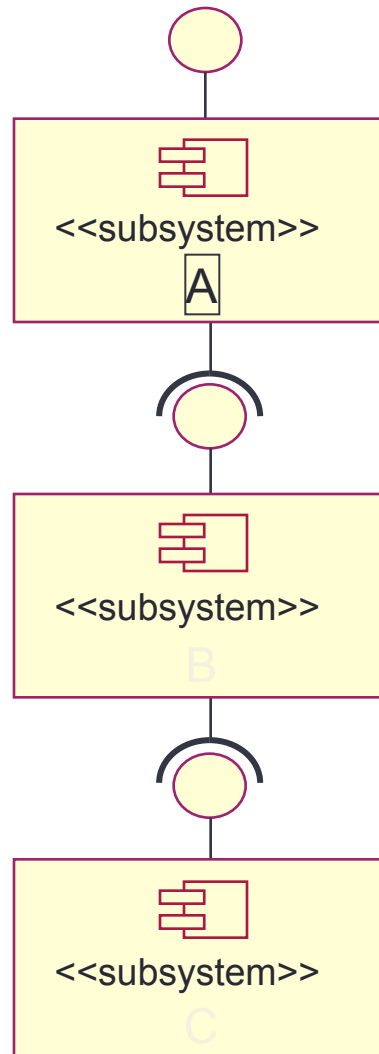


Diagrama de Componentes UML 2.0



Componentes

A UML define cinco estereótipos-padrão que se aplicam aos componentes:

- 1) <<EXECUTÁVEL >>: que poderá ser executável.
- 2) <<BIBLIOTECA>>: biblioteca de objetos estática ou dinâmica.
- 3) <<TABLE>>: uma tabela de banco de dados.
- 4) <<FILE>>: um documento contendo código fonte ou dados.
- 5) <<DOCUMENTS>>: representa um documento.

Componentes

- Modelagem de executáveis e bibliotecas

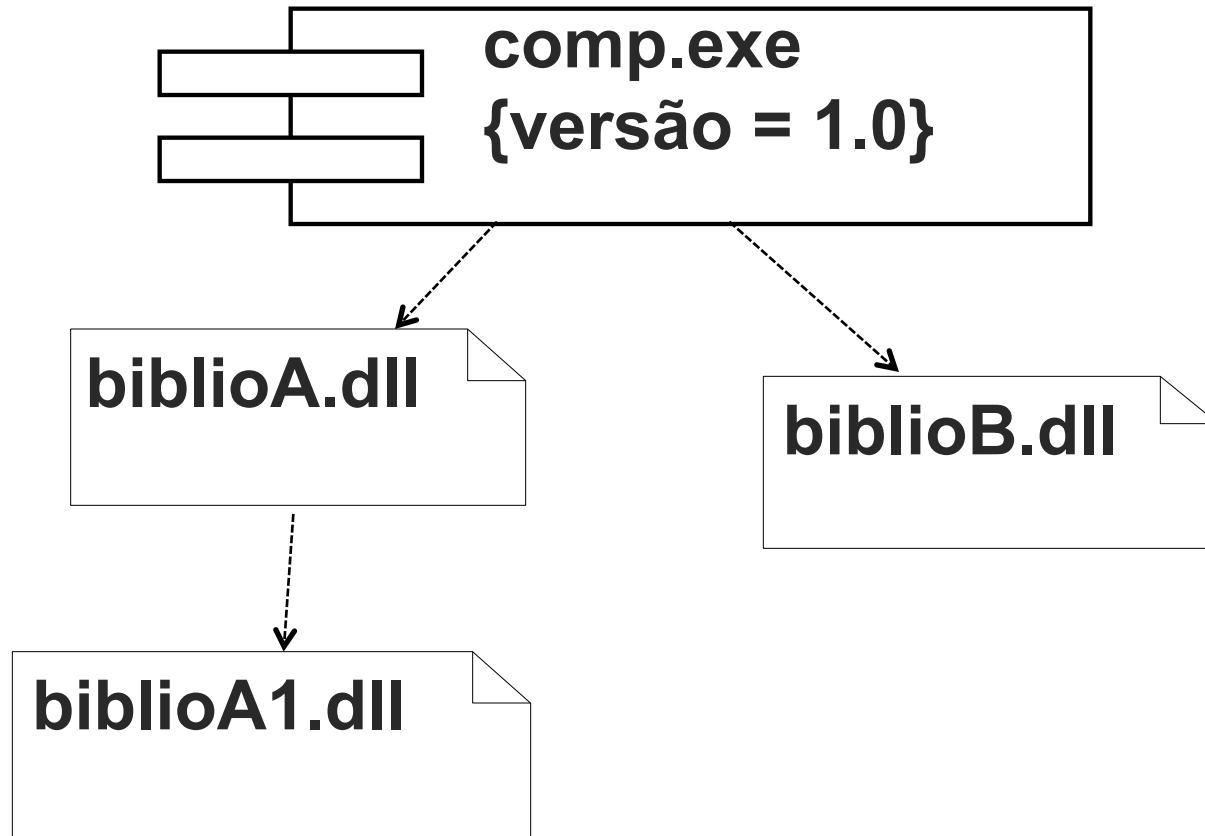
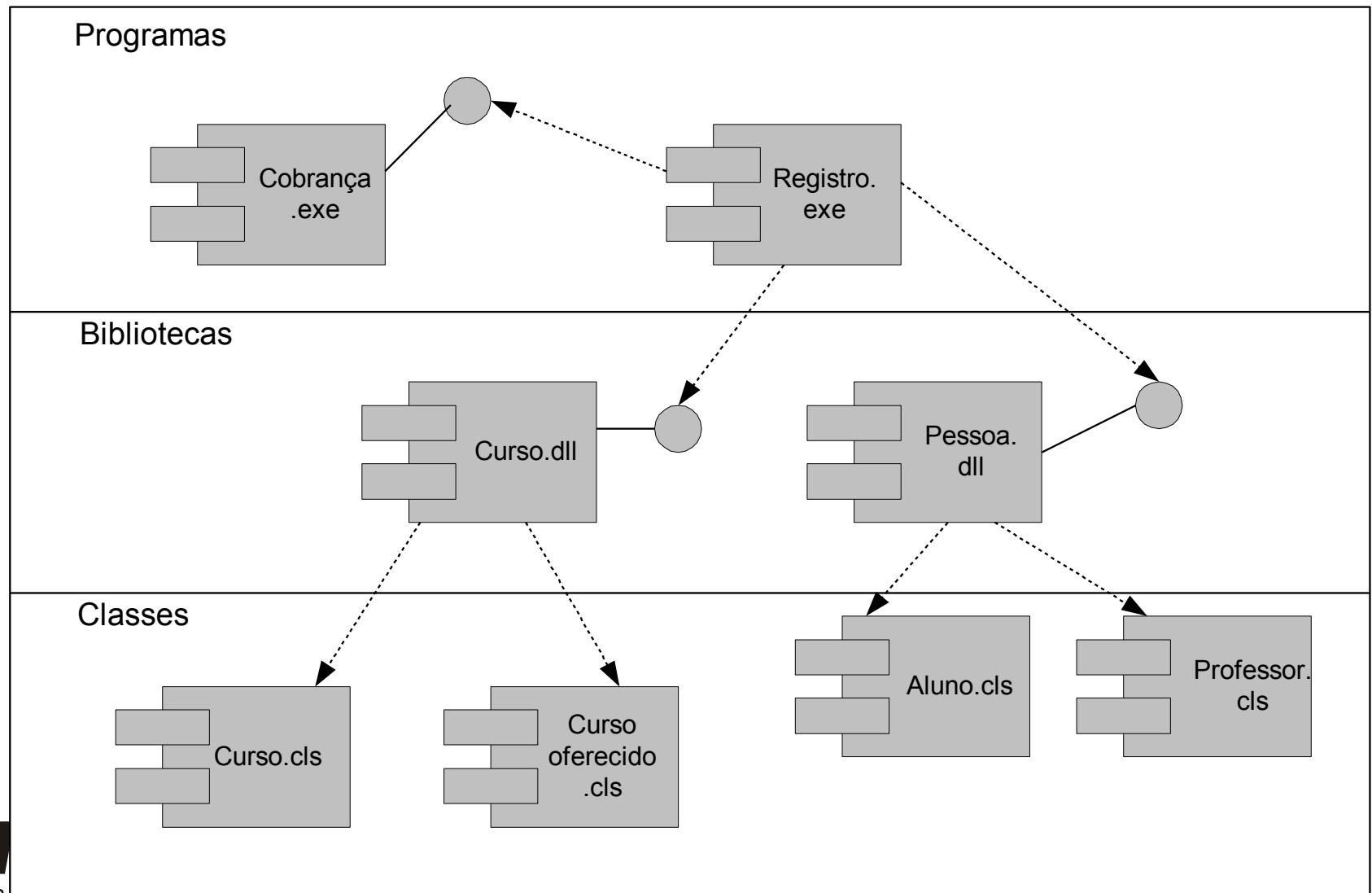
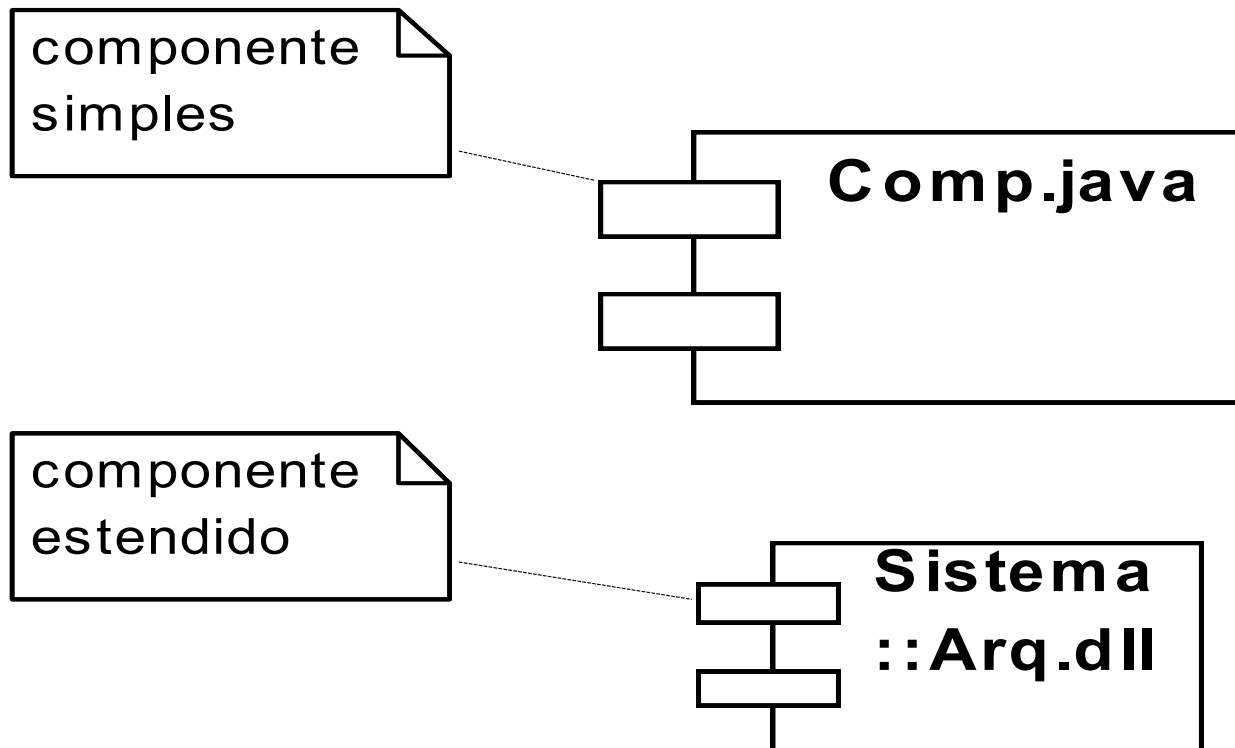


Diagrama de Componentes



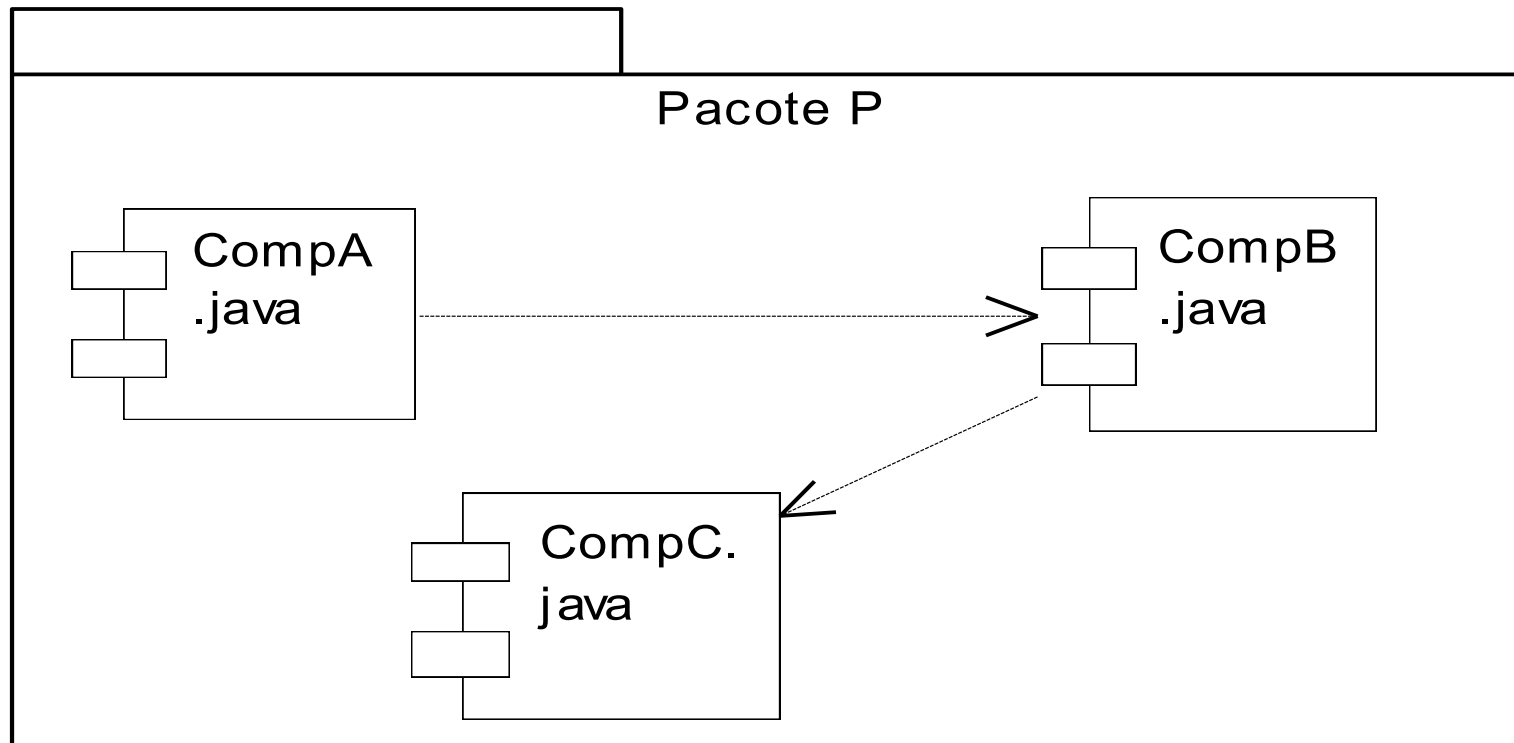
Componentes



O nome de um componente pode ser texto formado por qualquer quantidade de letras, números e certos sinais de pontuação (com exceção de sinais como dois-pontos, utilizado para separar o nome do componente e o nome do pacote que o contém).

Componentes

Ao fazer a modelagem de um sistema, a decisão de se utilizar uma classe ou um componente envolve uma escolha simples - se o que é modelado vive diretamente em um nó, use um componente; caso contrário, use uma classe.



Classes: Abstratas x Interfaces

Abstratas	Interfaces
<ul style="list-style-type: none">• Permitem declaração de atributos;• Mais ligadas a implementação;	<ul style="list-style-type: none">• Estão mais ligadas a comportamento

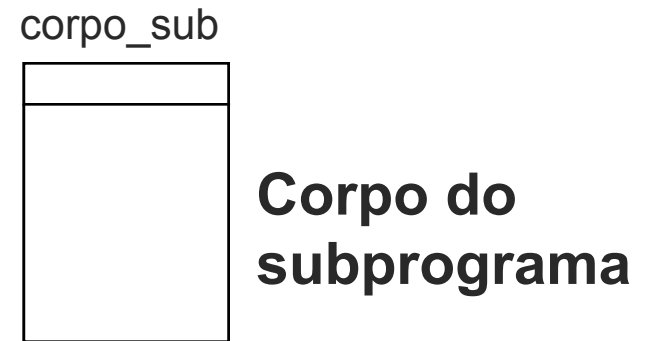
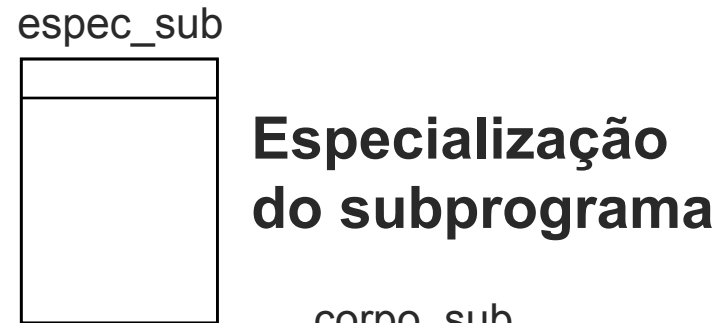
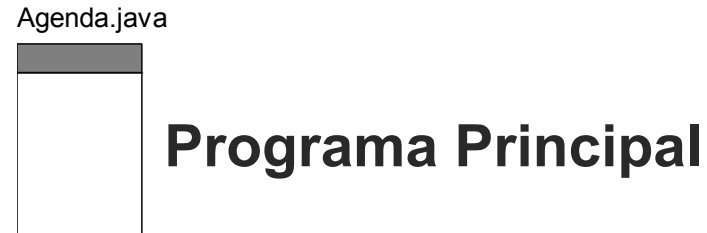
Os métodos definidos na interface são “OCOS” ou desprovidos de implementação. Classes podem dizer que implementam uma interface, estabelecendo um compromisso, uma espécie de contrato.

Componentes X Classes

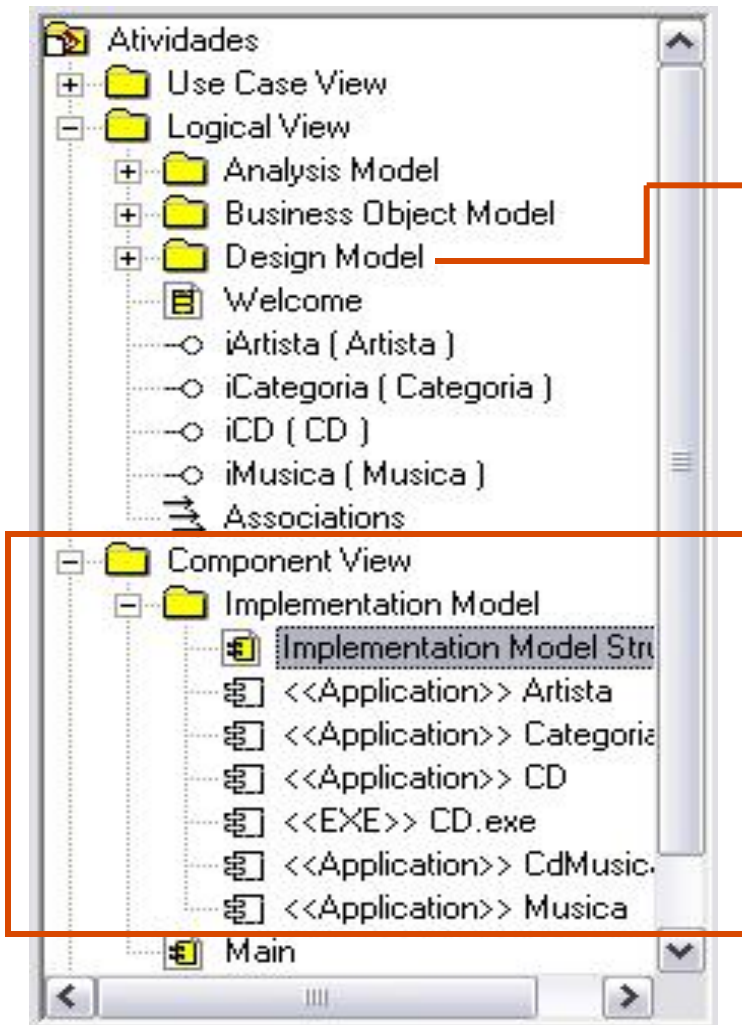
- | | |
|---|---|
| <ul style="list-style-type: none">• Representam coisas físicas que vivem no mundo dos bits.• Componentes podem viver em nós. As classes não.• Representam o pacote físico de componentes lógicos e se encontram em um nível diferente de abstração• Os componentes têm operações que são alcançadas por meio de suas interfaces. | <ul style="list-style-type: none">• Representam abstrações lógicas;• As classes podem ter atributos e operações diretamente. |
|---|---|

Diagrama de Componentes

➔ É composto por:



Construção do Diagrama de Componentes utilizando o Rose



• Utilize o ícone  para criar as interfaces

• Crie as interfaces dentro de:

Logical View : Design Model: (pasta Interfaces)

• Crie os componentes dentro de:

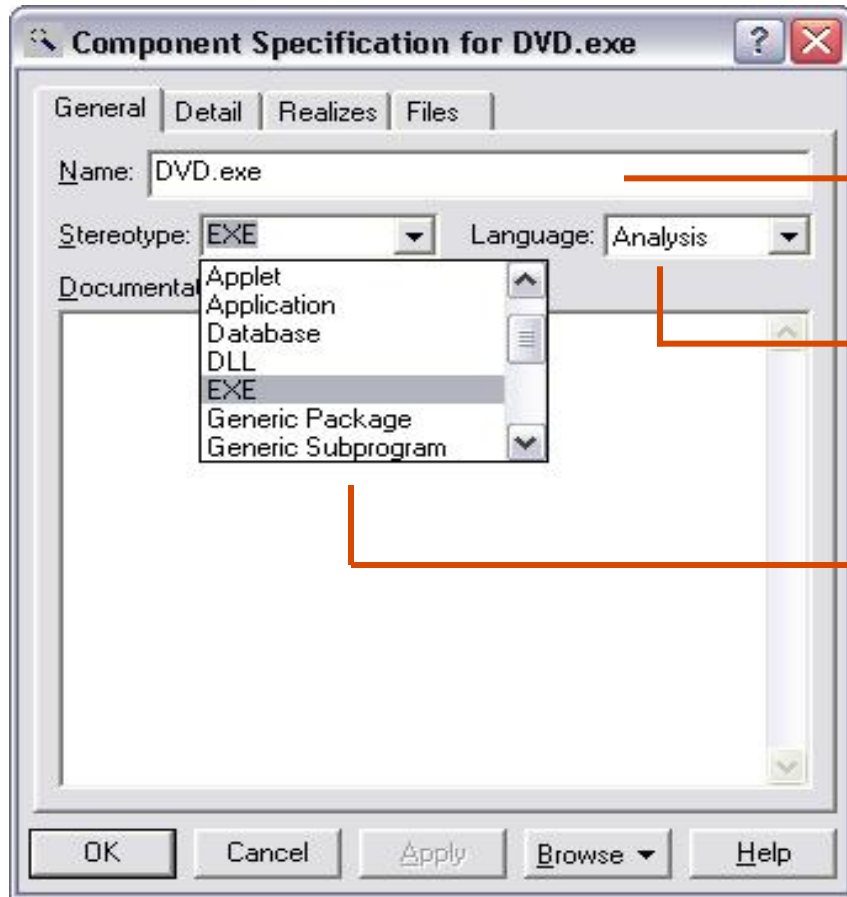
Component View

└─ *Implementation Model*

└─ *Implementation Model Structure*

Utilize o ícone  para criar os componentes

DIAGRAMA DE COMPONENTES



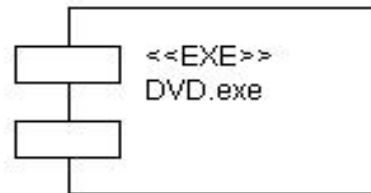
• Clique com MBD → *Open Specification* para:

• Adicionar nome ao componente

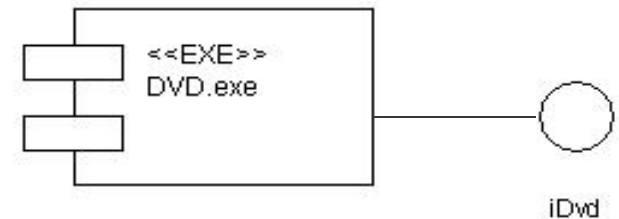
• Informar o Estereótipo do componente

• Informar a Linguagem

• Arraste as interfaces para dentro de seus respectivos componentes em *Implementation Model Structure*



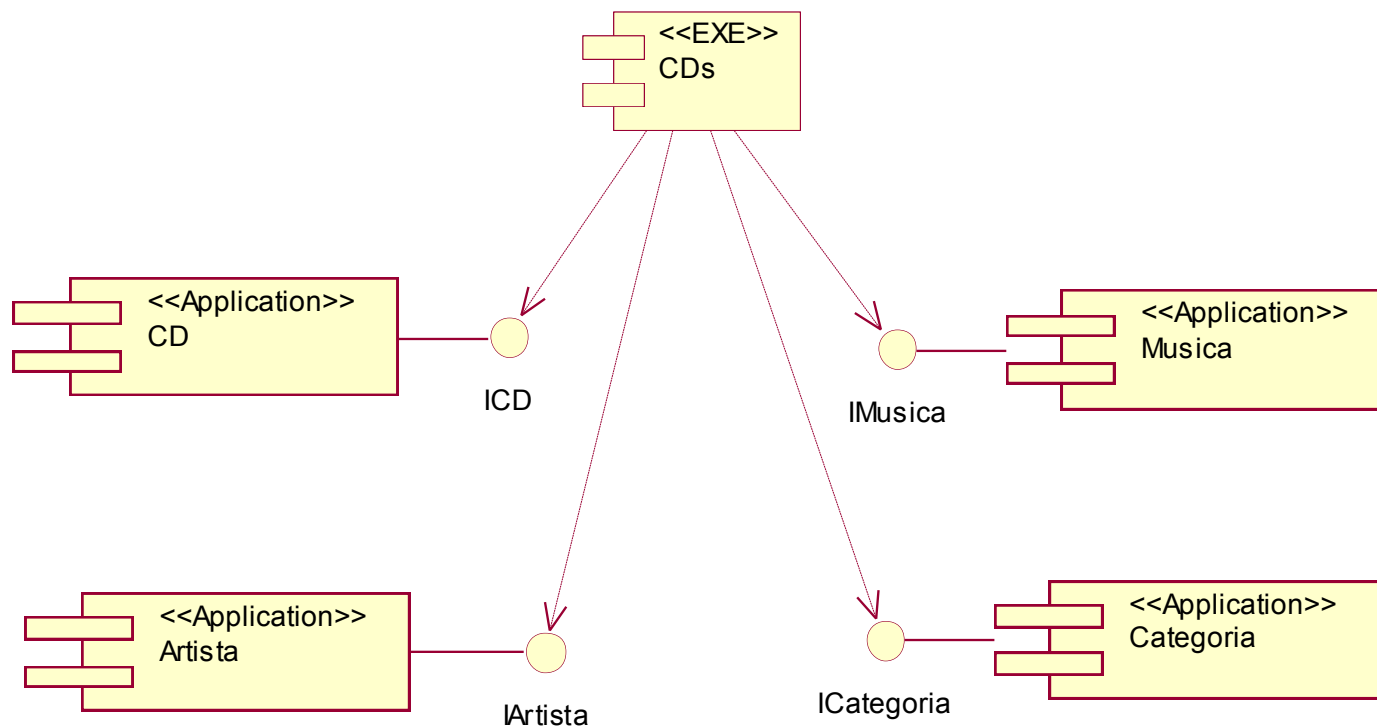
COMPONENTE



DEPOIS DE ARRASTAR A INTERFACE



INTERFACE



Diagramas de Implantação (instalação)

Arquitetura da Aplicação

- Qual hardware será necessário para executar a sua aplicação ?
- Sua aplicação será executada em uma máquina apenas ou em várias máquinas ?
- Qual (is) plataformas?
- Será necessário modularizar sua aplicação ?
- Qual banco de dados será utilizado?

Arquitetura da Aplicação

Recomenda-se:



1º - Decidir se a aplicação será executada em uma única máquina ou em múltiplas máquinas;

2º - Decidir o tipo de plataforma necessária.



3º - Para aplicações mais complexas construir protótipos para definir a interface do usuário/máquina.

Arquitetura da Aplicação

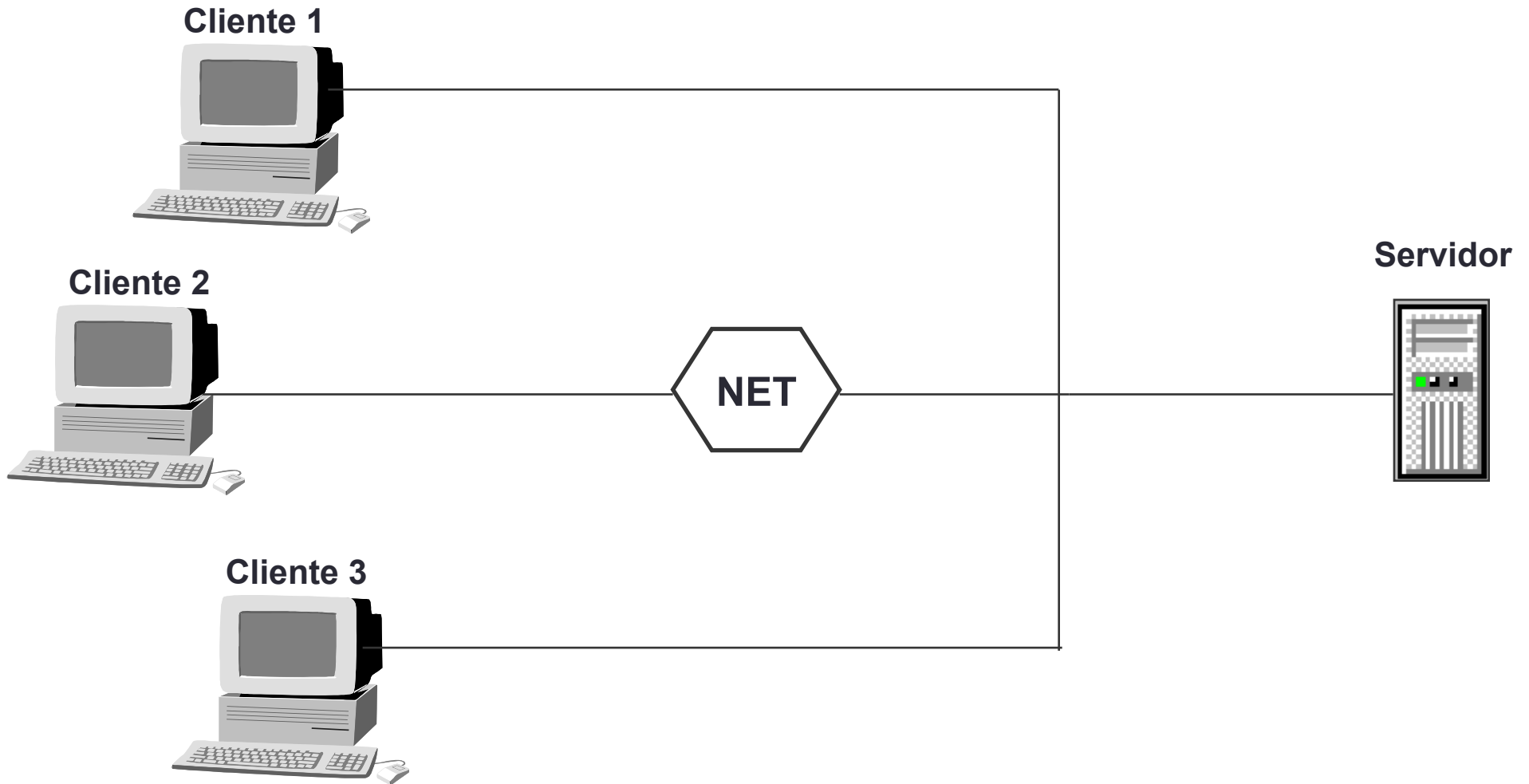


Diagrama de Implantação

- ▶ Mostra elementos de configuração de processamento *run time* e os componentes de software, processos e objetos que neles se mantêm;
- ▶ Inclui o uso físico do sistema (computadores, dispositivos);
- ▶ Símbolo de multiplicidade (*) pode ser colocado em nós para refletir o número de dispositivos;
- ▶ Um nó é desenhado como uma figura que apresenta visão tridimensional de um cubo.

Diagrama de Implantação

- Cada Nódulo (Nó) ou Processo é representado por uma caixa tridimensional;
- Cada Nódulo deve ser nomeado;
- As ligações entre os Nódulos são representadas por linhas.

◆ **Notação:**

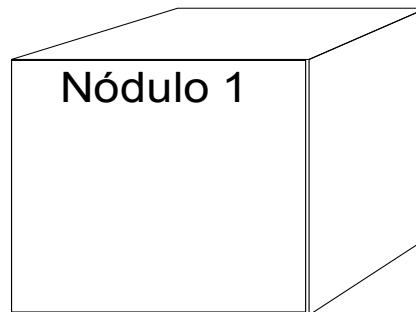


Diagrama de Implantação

◆ Notação de Ligação:

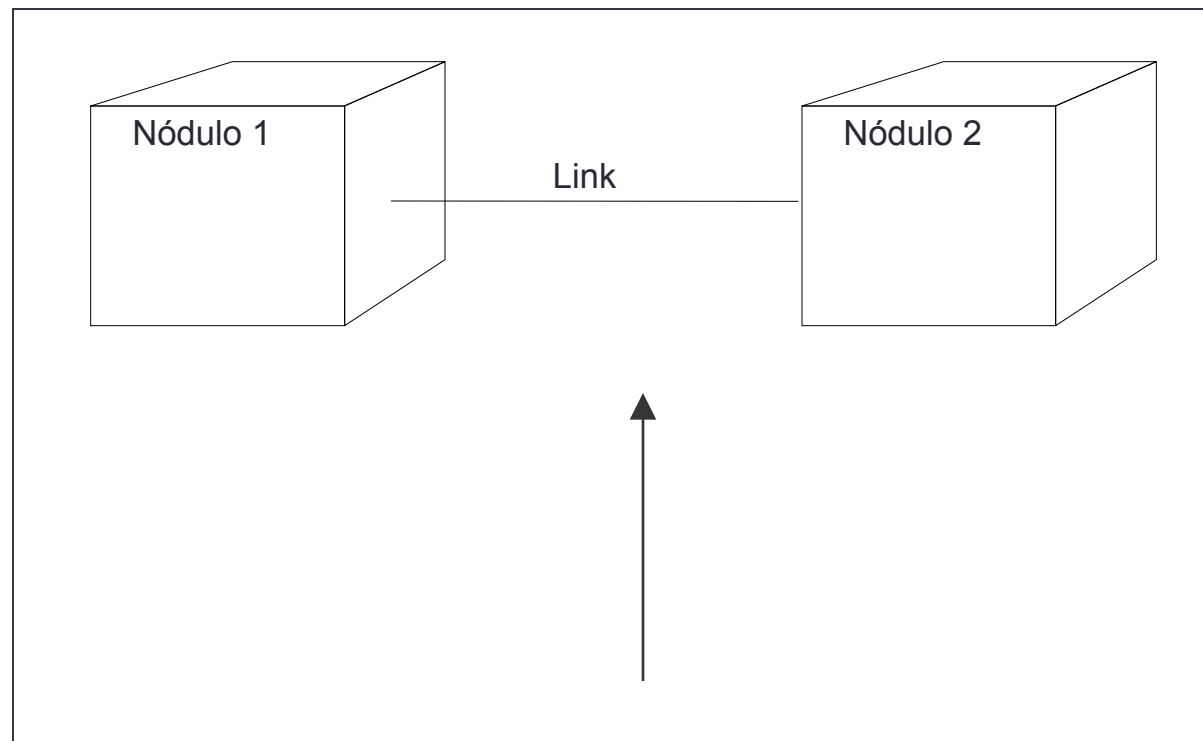


Diagrama de Implantação

◆ Exemplo:

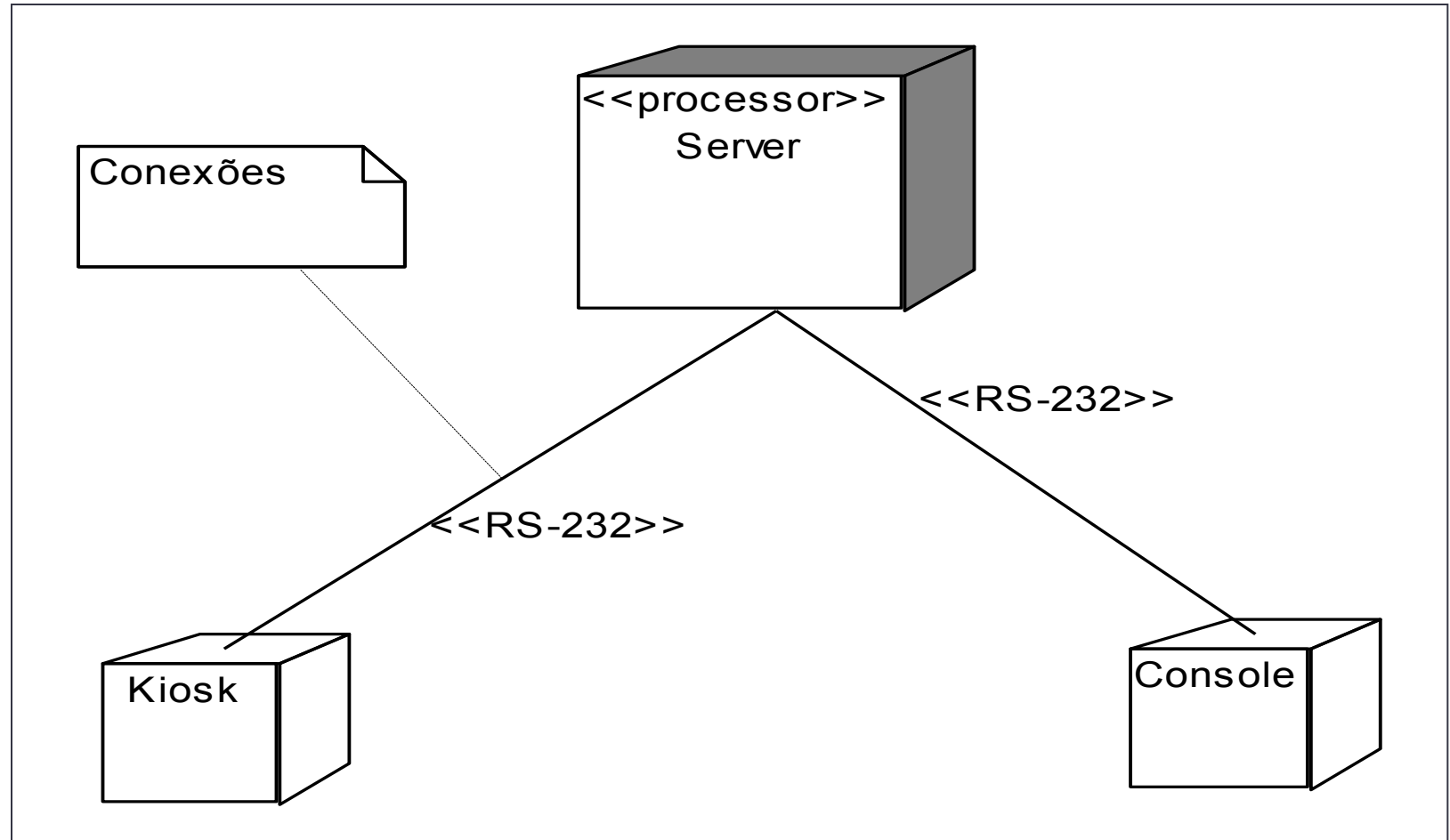


Diagrama de Implantação

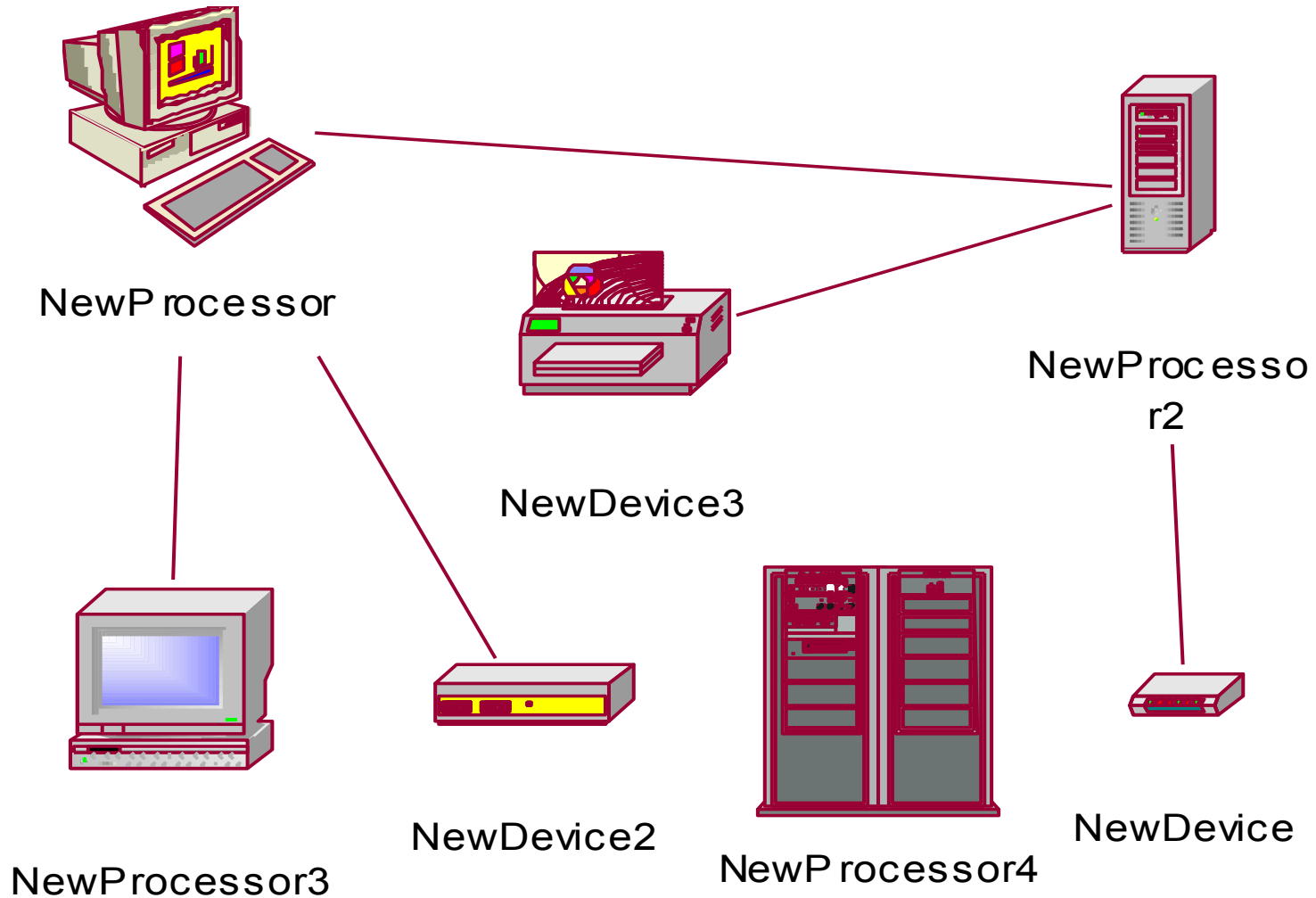


DIAGRAMA DE IMPLANTAÇÃO

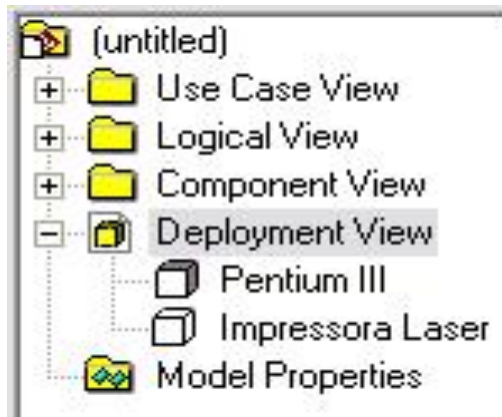
Exercício 08:

“Modele o ambiente no qual o Sistema de Controle de CD deverá ser executado, de acordo com a descrição a seguir:

*A empresa XYZ possui 03 estações de trabalho ligadas em rede através de um **HUB**. A **Estação I** pertence ao funcionário responsável pelo controle dos CDs. Já a **Estação II** pertence ao diretor geral da empresa, o qual possui uma **impressora Deskjet 3550 HP** para uso exclusivo. A **Estação III** é usada pela secretária apenas para preparação de documentos e outras atividades da empresa.*

*Além destas estações, o HUB encontra-se conectado, também, em um **Athlon XP 2.2 MHz (Servidor)**. A outra **impressora (HP Laser 2100)** está ligada ao servidor, bem como o modem ADSL para acesso à internet, sendo assim, ambos compartilhados (Impressora e modem) pelas 3 estações de trabalho da empresa.”*

Construindo o Diagrama de Implantação no Rose



- Dê duplo clique em “*Deployment View*”
- Crie o ambiente onde rodará o sistema com:
 - Processadores
 - Dispositivos
 - Links (Conexões)

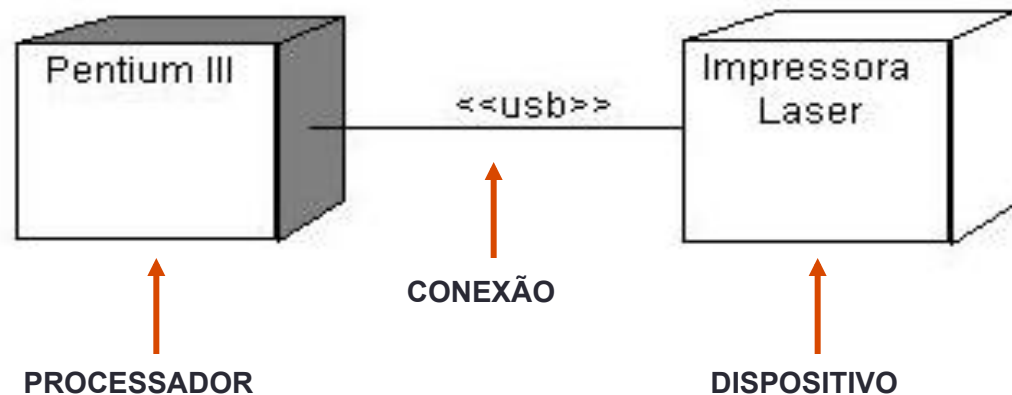


DIAGRAMA DE IMPLANTAÇÃO

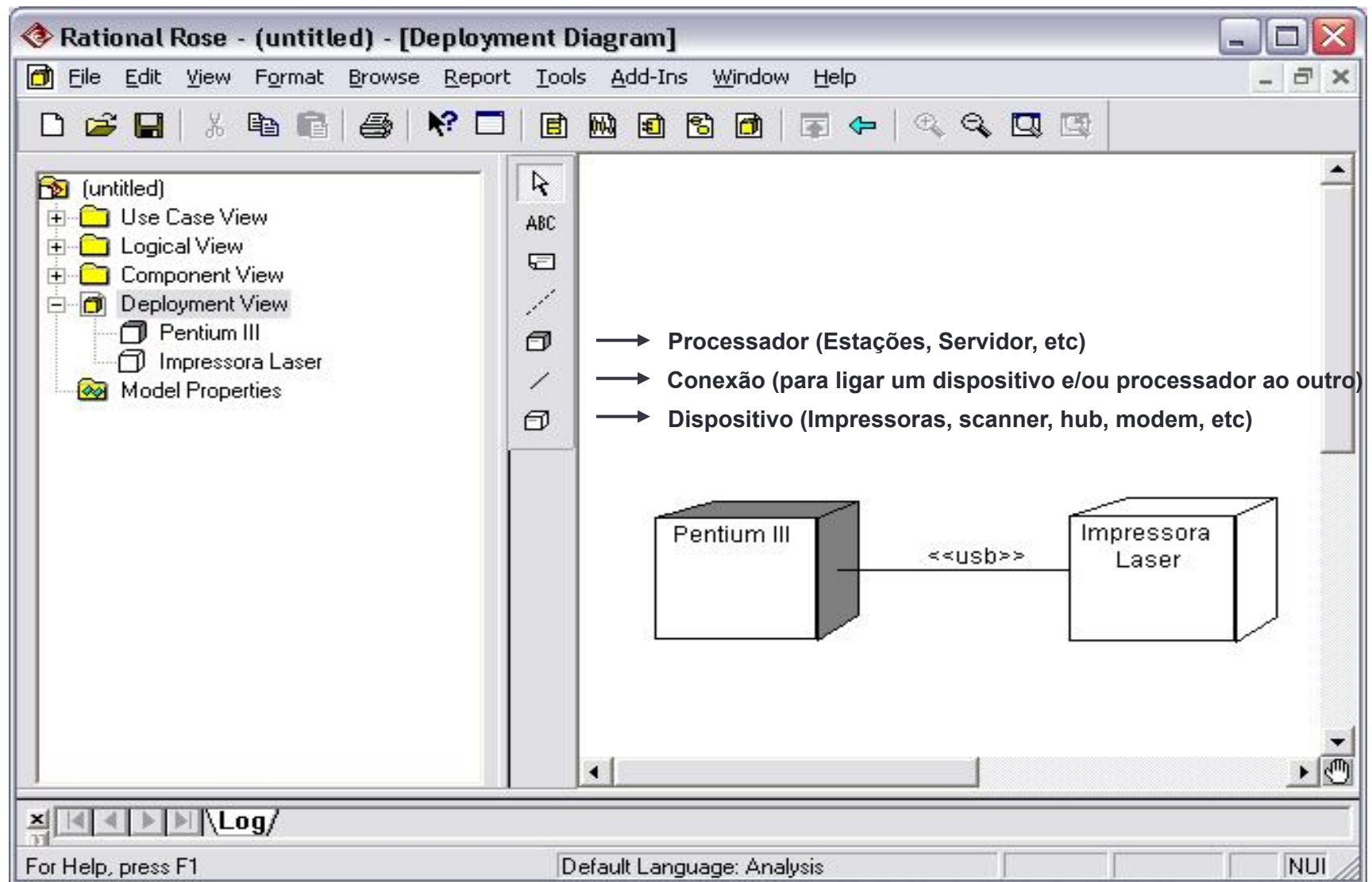
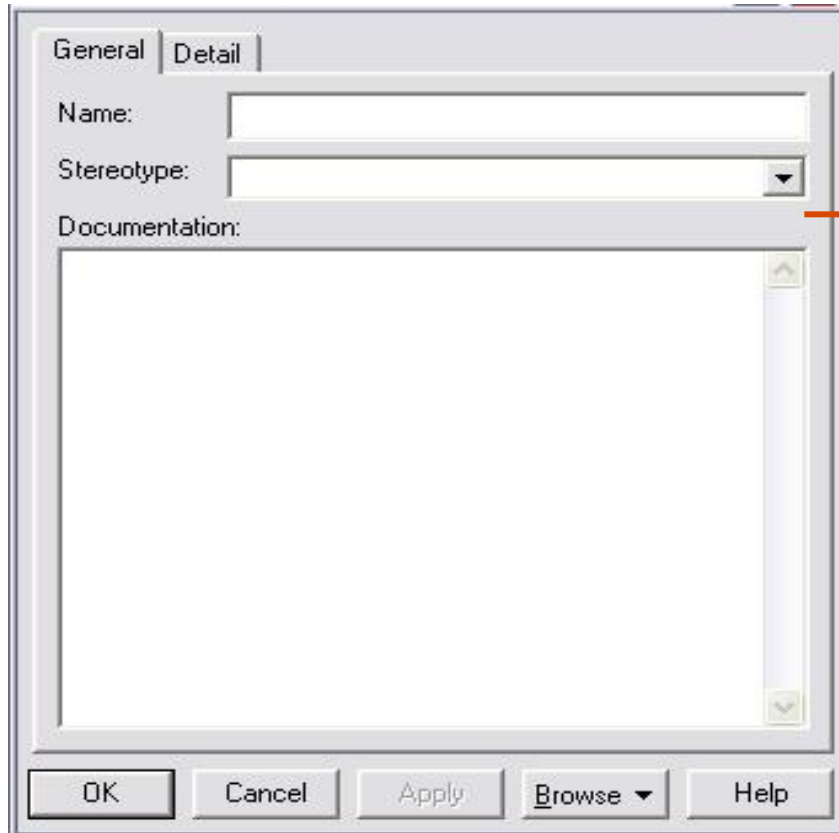


DIAGRAMA DE IMPLANTAÇÃO



- “Dê duplo clique em um processador e/ou dispositivo
- Selecione ou digite um Estereótipo para cada um

• Ex. usb

Impressora

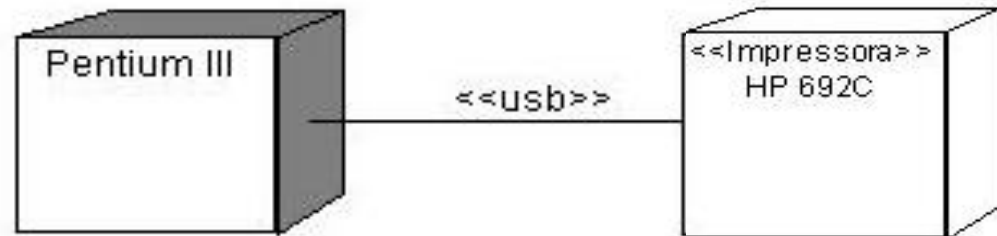


DIAGRAMA DE IMPLANTAÇÃO

