

# Engenharia de Software II



- **Diagrama de Componentes**
- **Diagrama de Implantação**

# Engenharia de Software II

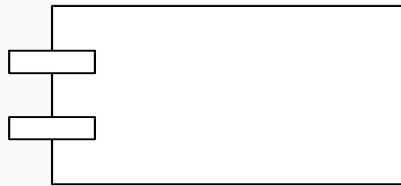
## Diagrama de Componentes

Este diagrama apresenta uma versão estática do sistema:  
Quais os módulos que o compõem.  
Está ligado diretamente à linguagem de programação a ser  
utilizada na implementação do sistema.  
Representa quais os arquivos que compõem o sistema:  
códigos-fonte, bibliotecas, tabelas, arquivos de configuração,  
formulários, relatórios e outros.  
Utilizado em engenharia reversa, partindo-se deste para a  
dissecção do sistema em módulos.

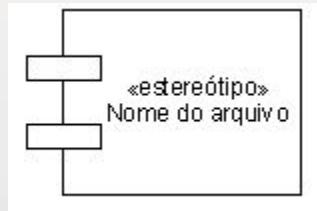
# Engenharia de Software II

## Diagrama de Componentes

Cada arquivo é considerado um componente, sendo representado por este símbolo:



Dentro do símbolo descrevemos o nome do componente e, opcionalmente, um estereótipo, indicando a finalidade do arquivo. Existem alguns estereótipos comuns, mas você pode criar outros.



# Engenharia de Software II

## Estereótipos

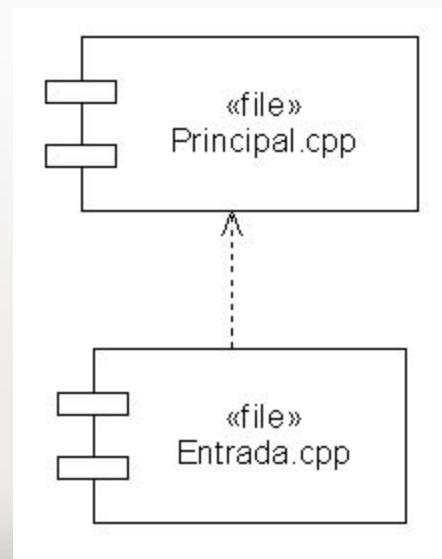
### Estereótipos comuns:

- **<<executable>>**: arquivos executáveis, já compilados e linkeditados;
- **<<library>>**: bibliotecas contendo funções, classes e procedures;
- **<<table>>**: tabelas (de banco de dados), repositório físico dos dados;
- **<<document>>**: arquivos textos (configuração, dados, etc.);
- **<<file>>**: qualquer outro tipo de arquivo, como código-fonte por exemplo.

# Engenharia de Software II

## Dependência

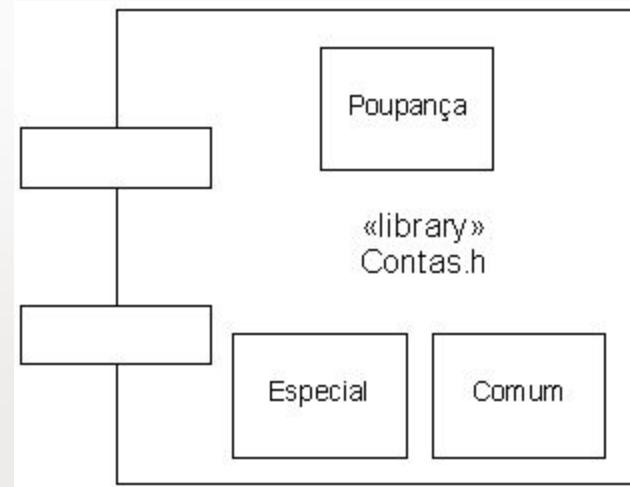
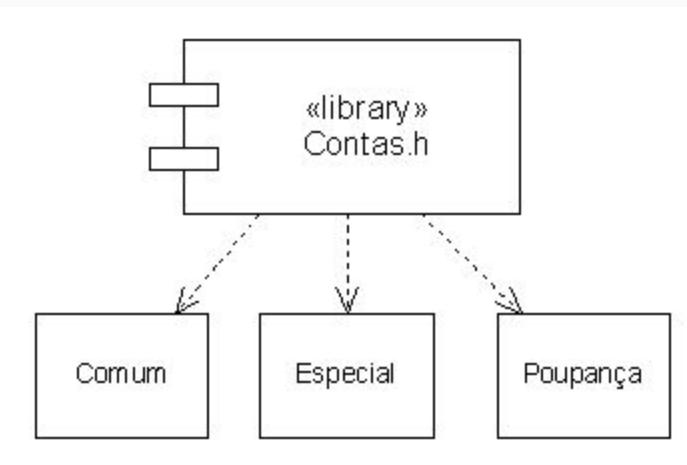
Uma seta indica que um componente está subordinado à outro.  
A isto chamamos de relacionamento de dependência



# Engenharia de Software II

## Dependência

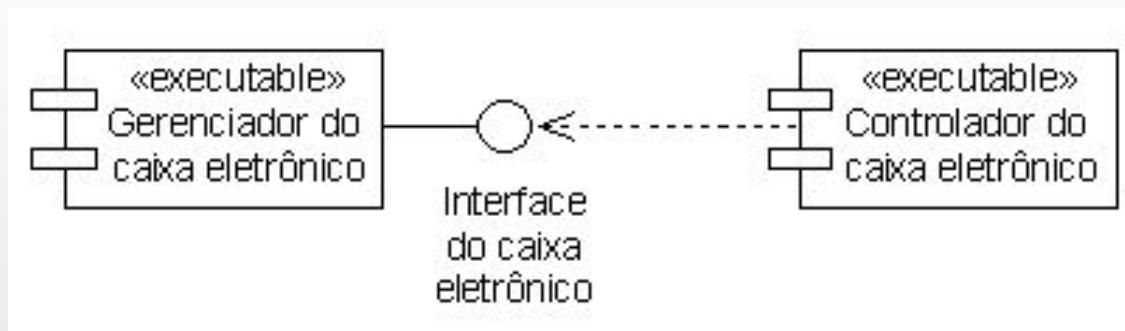
Pode indicar ainda classes pertencentes à determinado componente:



# Engenharia de Software II

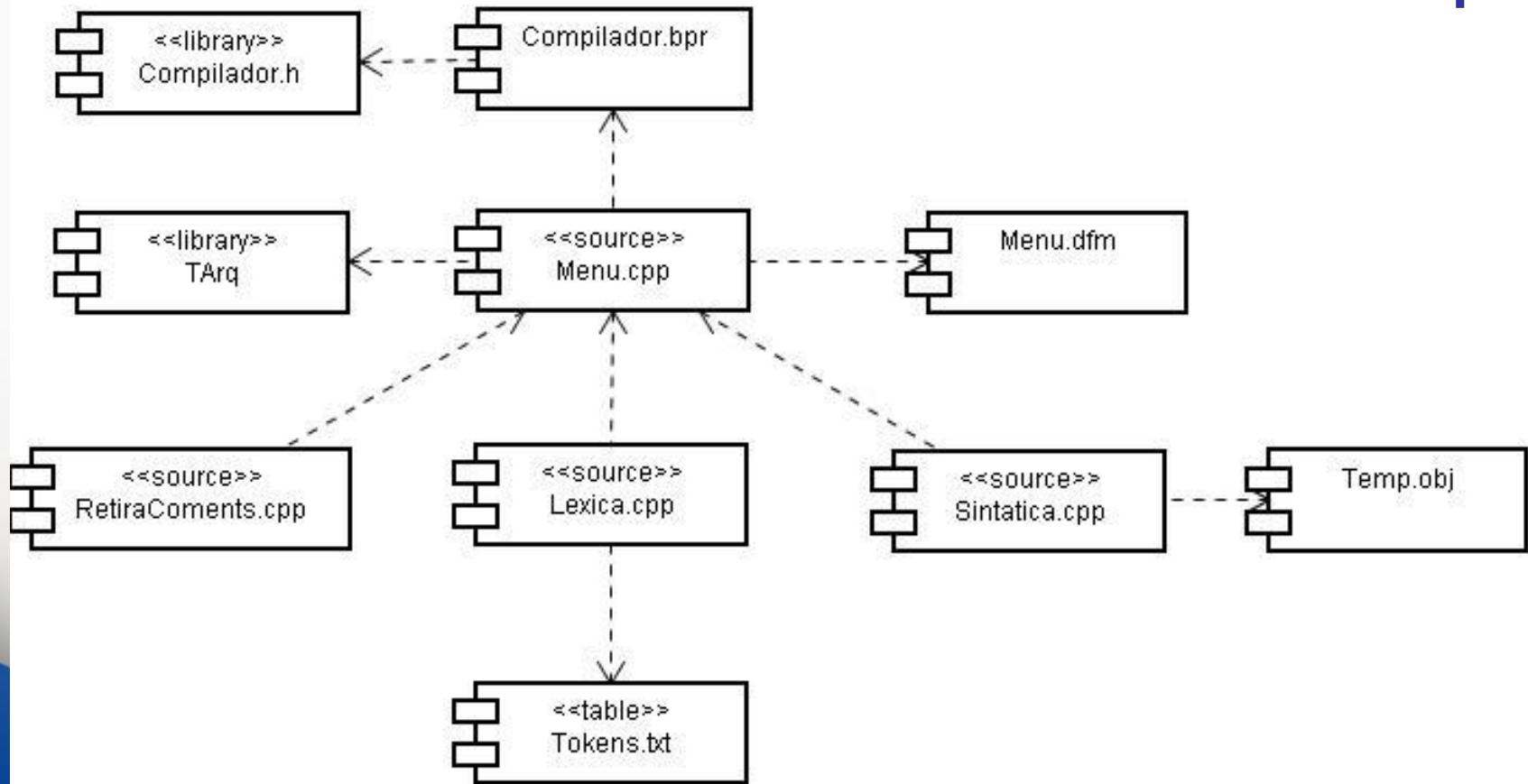
## Interface

Representa um procedimento realizado por uma classe ou componente que envolve uma interface (vide diagrama de colaboração).



# Engenharia de Software II

## Exemplo



# Engenharia de Software II

Material de apoio:

[https://sparxsystems.com/enterprise\\_architect\\_user\\_guide/17.1/modelling\\_languages/componentdiagram.html](https://sparxsystems.com/enterprise_architect_user_guide/17.1/modelling_languages/componentdiagram.html)

# Engenharia de Software II

## Diagrama de Implantação

Este diagrama evidencia a instalação física do sistema: servidores, protocolos, estações, etc.), além de definir como os componentes físicos desta instalação irão se comunicar.

Este diagrama só tem utilidade quando o sistema será particionado entre diversos equipamentos: não tem sentido se rodar em apenas uma máquina.

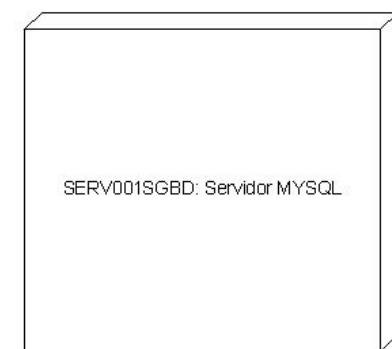
# Engenharia de Software II

## Nós

Um nó (representado pela figura abaixo) representa uma máquina (computador) onde parte (ou partes) do sistema será executada, ou onde alguns arquivos serão armazenados.

O nó pode conter o nome de identificação da máquina em uma rede.

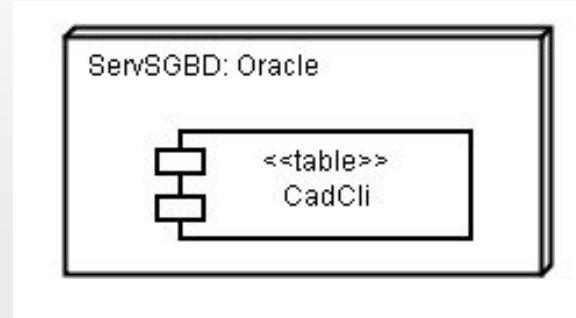
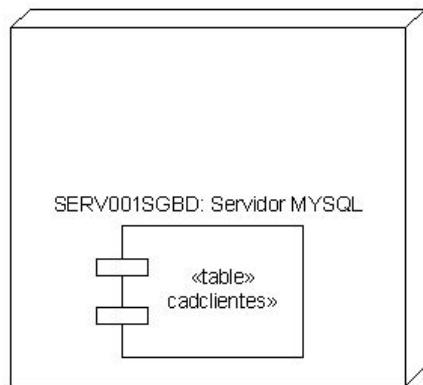
Pode também representar “máquinas virtuais” (diretórios, unidades de rede mapeadas, emuladores).



# Engenharia de Software II

## Nós com componentes

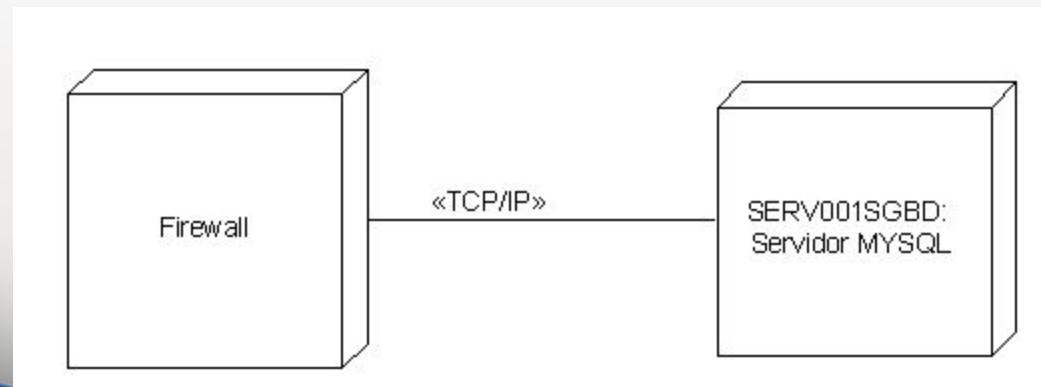
Pode-se identificar quais componentes estão “abrigados” em um determinado nó, bem como onde serão executados determinados módulos do sistema.



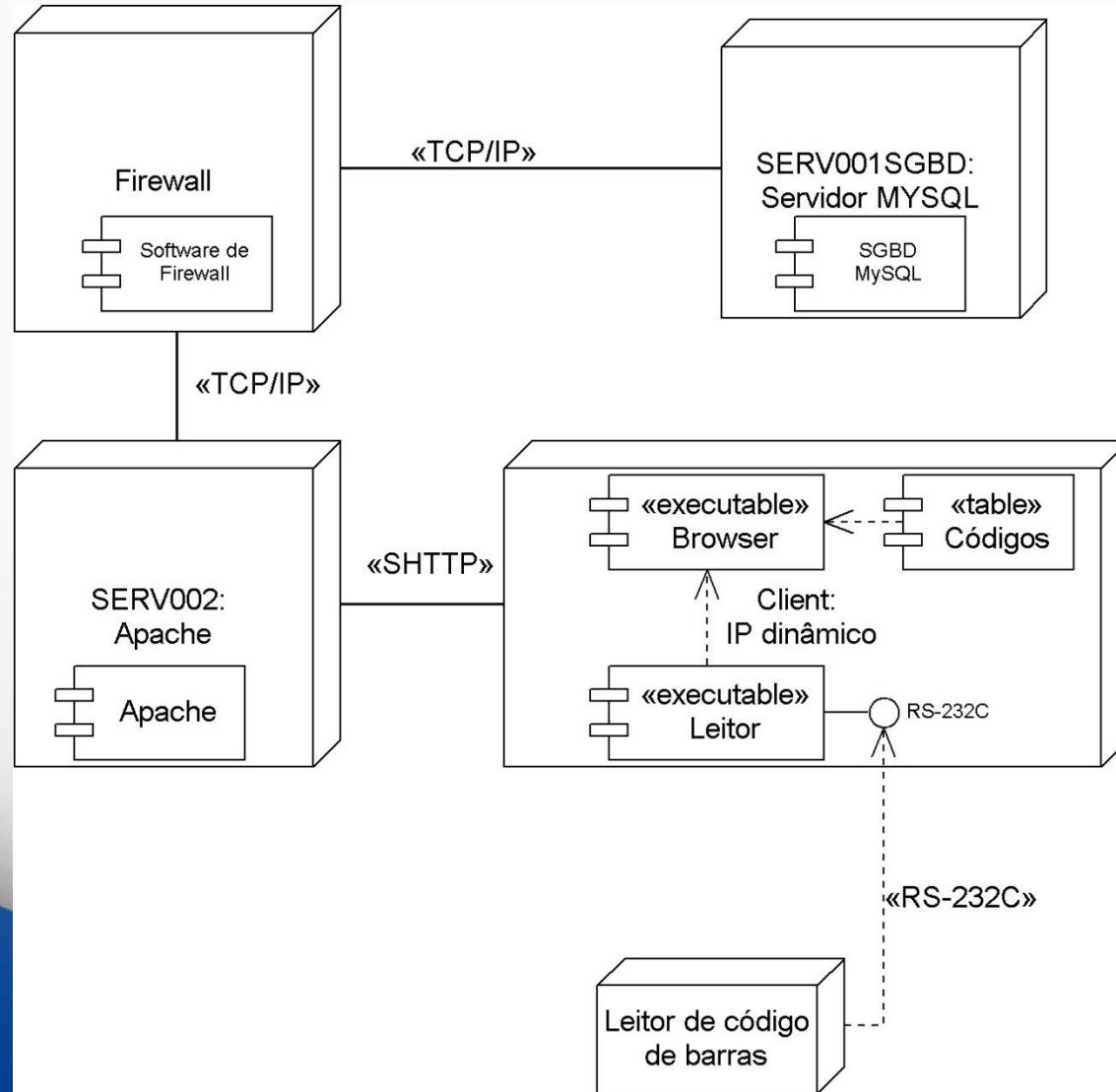
# Engenharia de Software II

## Associações

Os Nós devem trocar informações entre si através de alguma conexão física. A associação (representada por uma reta) descreve a associação física entre os nós, bem como – opcionalmente – o protocolo entre eles



# Engenharia de Software II



Exemplo

# Engenharia de Software II

Material de apoio:

<https://sparxsystems.com/resources/tutorials/uml2/deployment-diagram.html>