

Módulo 13

Socket

Programação Orientada a Objetos

Java

(Rone Ilídio)

O que é socket?

- **Definição:**

Link de comunicação entre dois processos que podem estar em computadores distintos.

- **Utilidade:**

Comunicação entre máquinas conectadas em uma rede.

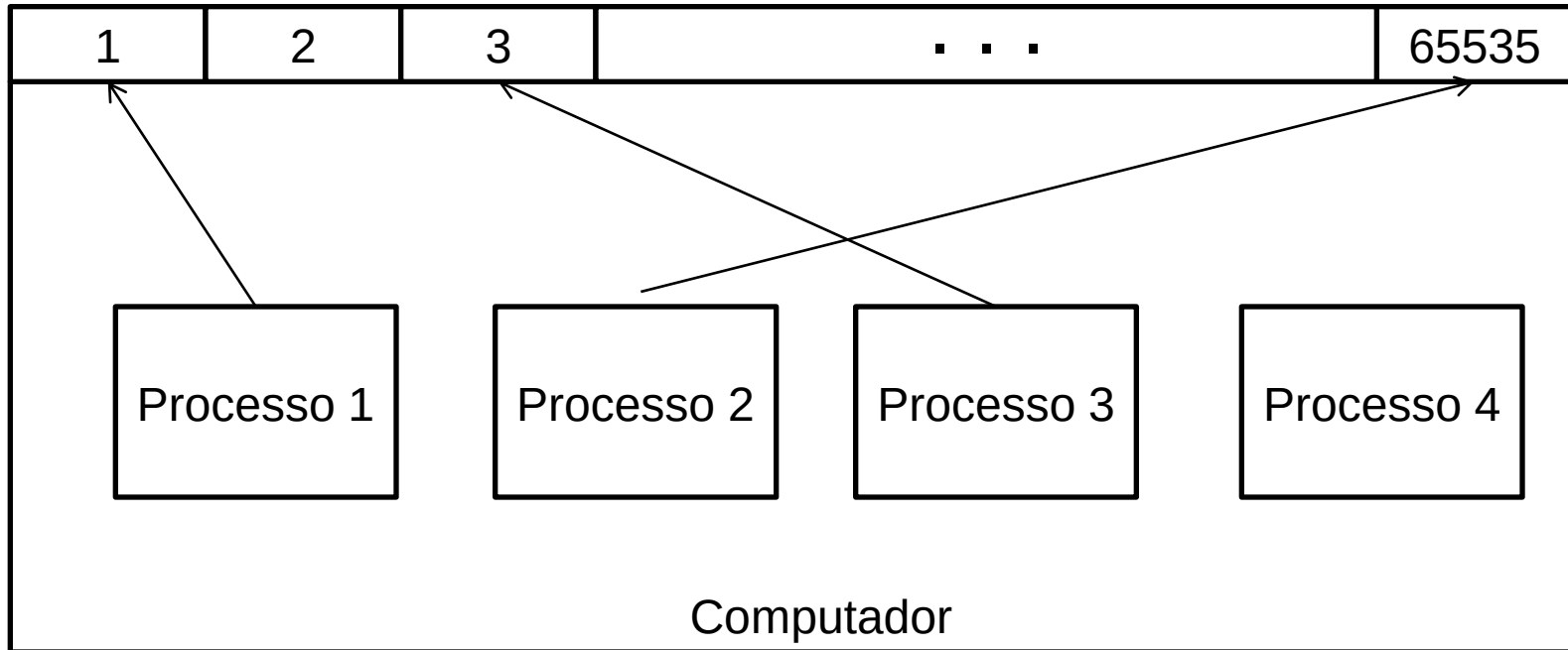
Endereço IP

- Número que identifica todos os computadores conectados à internet
- Exemplo: 255.255.255.255 (obs: ipv4)
- Como saber o ip no Windows → no prompt de comandos digite “ipconfig”



Portas

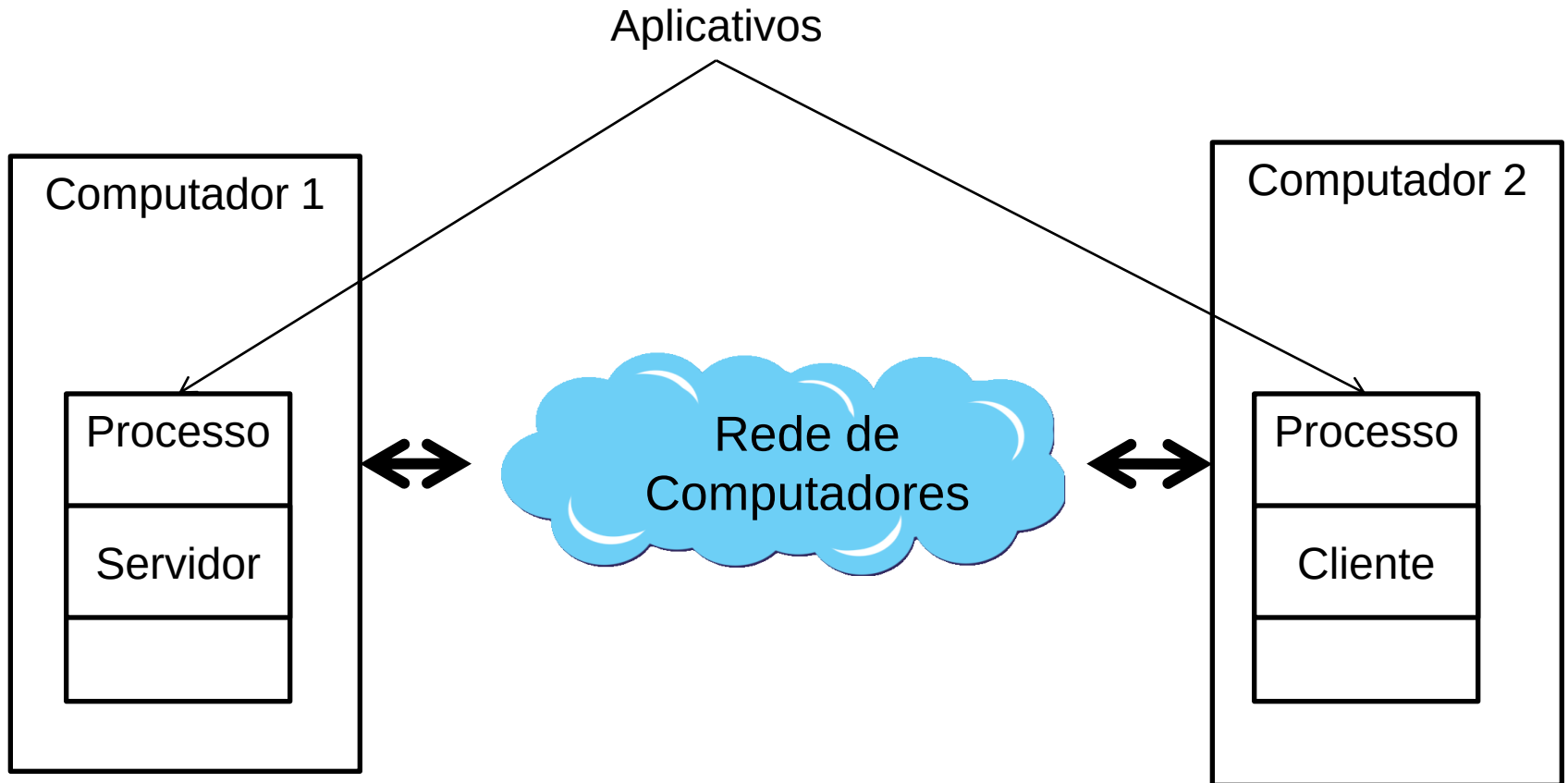
- Número de 16 bits que identifica o processo
- Canais por onde as aplicações se comunicam
- Portas de 0 a 1023: reservadas



Class Importantes

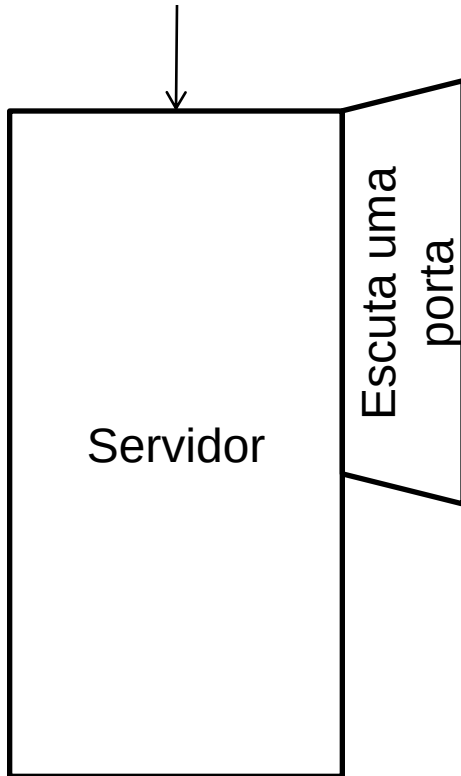
- Pacote java.net
- **ServerSocket**: define objetos servidores, os quais abrirão um canal de comunicação e esperarão conexões externas
- **Socket**: define objetos que formarão o link (um no cliente outro no servidor)
- Obs: a conexão não depende da linguagem de programação

Exemplo de conexão

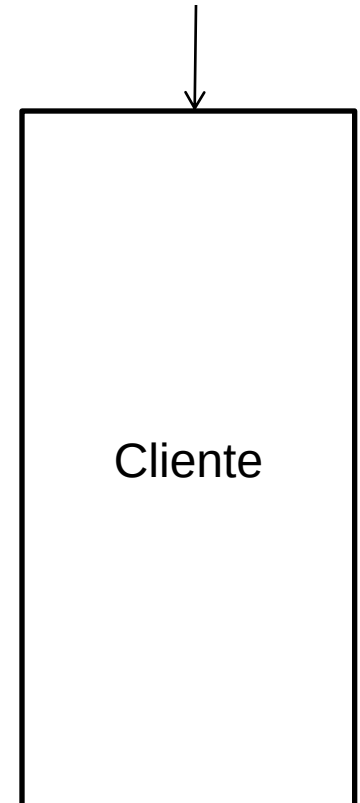


Exemplo de conexão

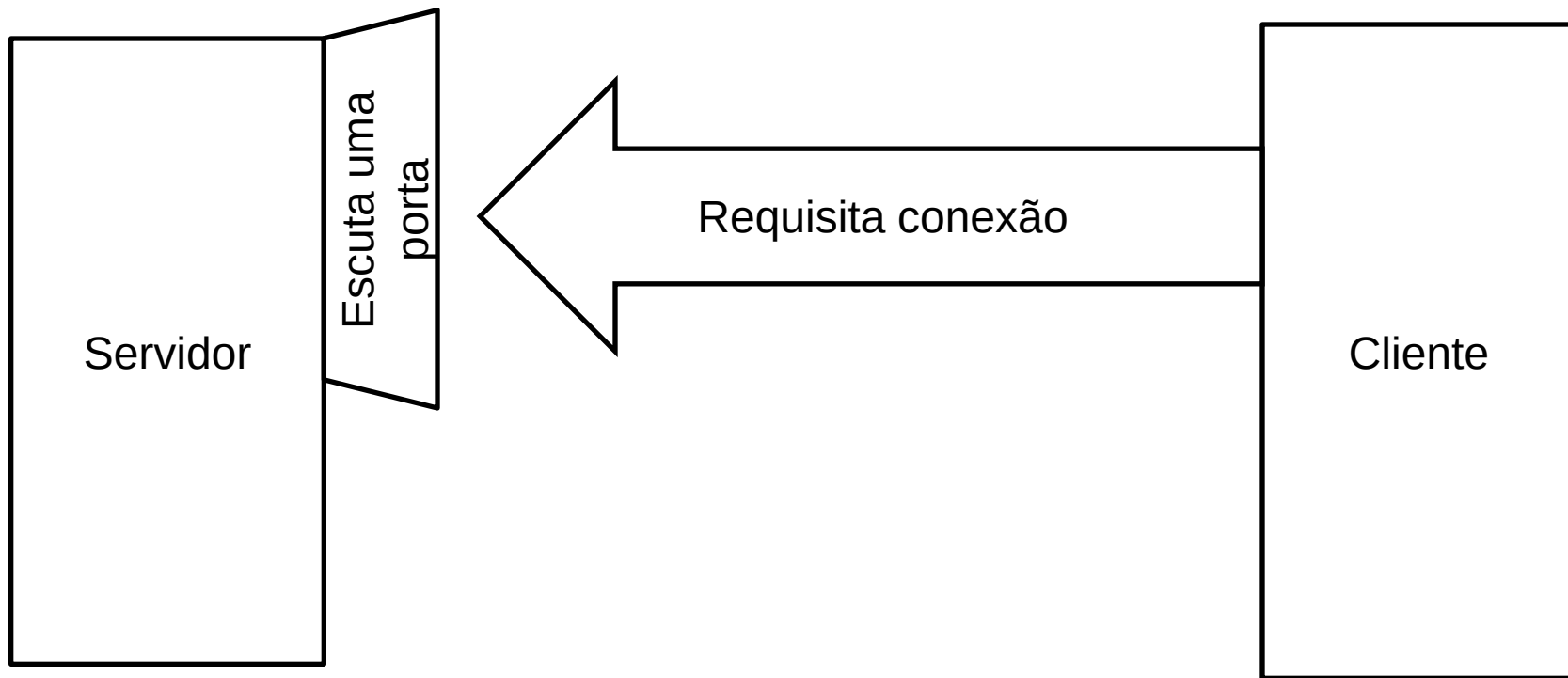
Objeto do tipo
ServerSocket



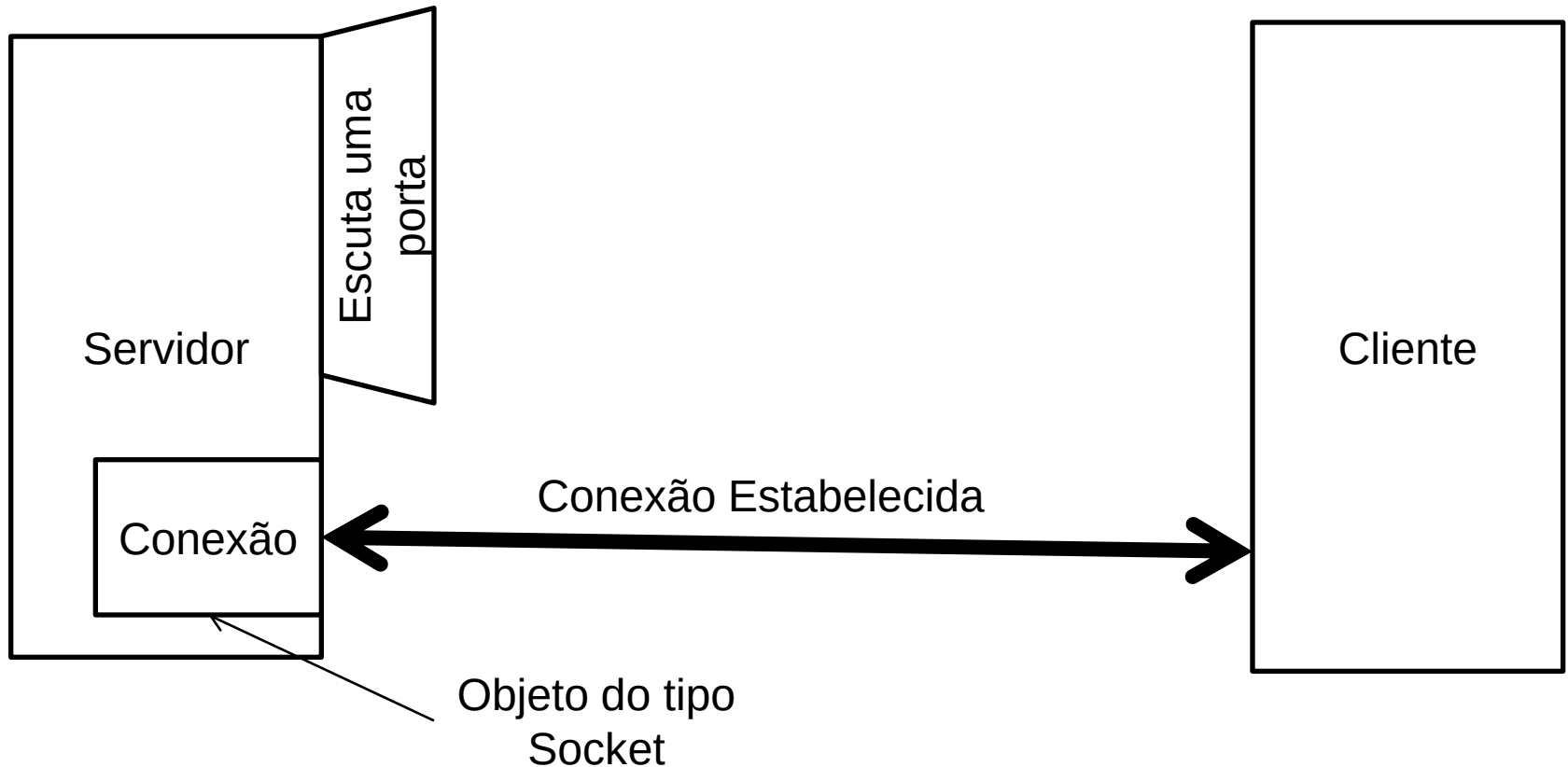
Objeto do tipo
Socket



Exemplo de conexão



Exemplo de conexão



Exemplo de Socket

- Sequência para criar um servidor
 - Criar um objeto do tipo `ServerSocket`
 - Fazer esse objeto escutar uma porta
 - Ao escutar uma conexão, criar um objeto do tipo `Socket`
 - Criar objetos de escrita e leitura e associar ao objeto `Socket`
- O servidor reenvia todas as mensagens recebidas.

```
import java.net.*; import java.io.*;
public class MyServer {
    public static void main(String args[]){
        ServerSocket socketServidor = null;
        try{
            socketServidor = new ServerSocket(5000);
        }
        catch(IOException e){
            System.out.println("Erro ao criar o ServerSocket");
            System.exit(1);
        }
        Socket socketCliente = null;
        try{
            socketCliente = socketServidor.accept();
        }
        catch(IOException e){
            System.out.println("Erro ao criar o ClientSocket");
            System.exit(1);
        }
    }
}
```

```
try{
    PrintWriter out = new PrintWriter(socketCliente.getOutputStream(), true);
    BufferedReader in = new BufferedReader(new
                                InputStreamReader(socketCliente.getInputStream()));

    String entrada;
    while((entrada = in.readLine()) != null){
        System.out.println("O Servidor recebeu:"+entrada);
        out.println(entrada);
        if(entrada.equals("fim")) break;
    }
    out.close();
    in.close();
    socketServidor.close();
    socketCliente.close();
}
catch(IOException e){
    System.out.println("Erro ao criar os streams");
    System.exit(1);
}
}
```

```

import java.io.*;
import java.net.*;
import javax.swing.JOptionPane;
public class MyClient {
    public static void main(String args[]){
        Socket socket = null;      //Socket cliente
        PrintWriter out = null;    //Escreve no socket
        BufferedReader in = null;  //Le do socket
        try{
            socket = new Socket("localhost",5000);
            out = new PrintWriter(socket.getOutputStream(),true);
            in = new BufferedReader(new InputStreamReader(socket.getInputStream()));

            String entradaservidor="";
            String entradausuario = JOptionPane.showInputDialog(
                "Digite para enviar para o servidor:");
            while(!entradausuario.equals("fim")){
                out.println(entradausuario);
                entradaservidor = in.readLine();
                entradausuario = JOptionPane.showInputDialog( "Chegou do servidor: "+
                    entradaservidor + "\nDigite para enviar para o servidor:");
            }
            out.close();
            in.close();
            socket.close();
        }
        catch(IOException e){
            System.err.println("Erro na criação dos objetos");
            System.exit(1);
        }
    }
}

```

```
try{
    socket = new Socket("localhost",5000);
    out = new PrintWriter(socket.getOutputStream(),true);
    in = new BufferedReader(new InputStreamReader(socket.getInputStream()));
    InputStreamReader(System.in));

    String entradaservidor="";
    String entradausuario = JOptionPane.showInputDialog(
                                                "Digite para enviar para o servidor:");
    while(!entradausuario.equals("fim")){
        out.println(entradausuario);
        entradaservidor = in.readLine();
        entradausuario = JOptionPane.showInputDialog( "Chegou do servidor: "+
                                                    entradaservidor + "\nDigite para enviar para o  servidor:");
    }
    out.close();
    in.close();
    socket.close();
}
```

```
catch(IOException e){
```

```
    System.err.println("Erro na criação dos objetos");
```

```
    System.exit(1);
```

```
}
```

```
}
```

```
}
```