

Redes de Computadores

Aulas Práticas

Material de suporte às aulas de Redes de Computadores – Copyright DI – FCT/ UNL / 1

Programação em Rede Baseada em Canais

Aula #3_

Material de suporte às aulas de Redes de Computadores – Copyright DI – FCT/ UNL / 2

Sumário

- Modelo Cliente/Servidor
- Comunicação com Canais
- Exemplo em Java
- Exercício (“ChatRoom”)
 - Concorrência no cliente...
 - Concorrência no servidor...

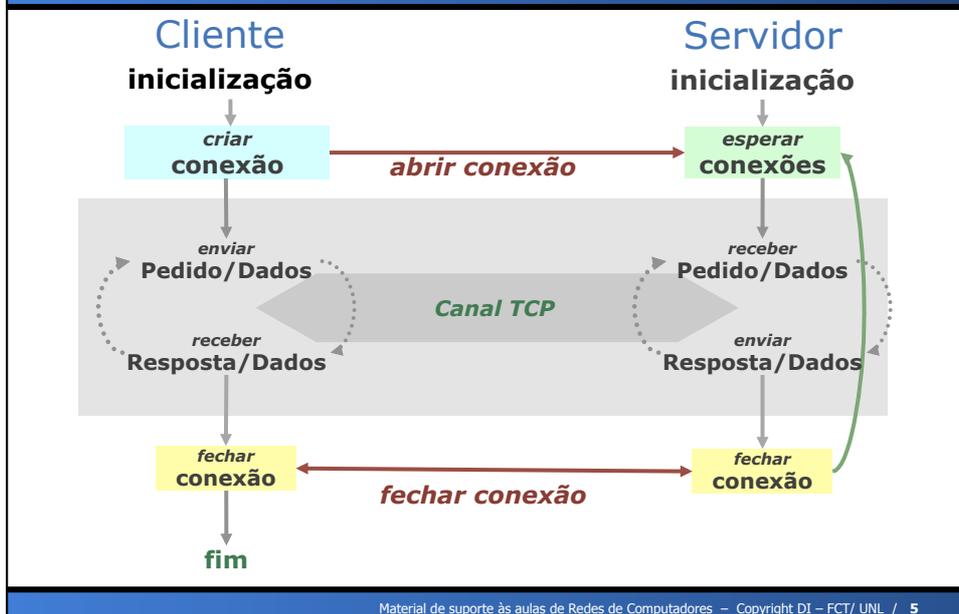
Modelo Cliente/Servidor

- **Duas** componentes autónomas:
 - Servidor, a **primeira** a correr e que geralmente **não termina**.
 - Cliente, normalmente invocada pelo **utilizador** para solicitar um serviço...

Portanto...

O modelo cliente/servidor continua a aplicar-se...

C/S diagrama de execução



Canal de Dados?

- Uma ligação **fiável** entre dois processos.
- **Iniciada pelo cliente** e endereçada a uma **máquina** e a um **porto**, os quais indicam onde o servidor aguarda os pedidos de conexão.
- Permite um fluxo **contínuo** e **bi-direccional** de dados (**bytes**) entre os dois processos.
- Pode ser **quebrada** por **qualquer um** dos participantes, em qualquer momento.
- Tanto o estabelecimento da ligação como o fecho da mesma é do conhecimento dos dois participantes...

Comunicação com Canais de Dados

- É feita por recurso a “sockets”, que neste caso estão diferenciados, segundo o seu papel:
 - Sockets TCP para aceitar ligações (servidor)
 - Sockets TCP para a troca de dados (cliente e servidor)
- Endereços que identificam a **origem** e o **destino** das conexões.
 - Máquina (Endereço IP)
 - 10.1.233.67, 127.0.0.1, 192.168.1.1, etc.
 - Porto (16 bits)
 - 8000, 80, 8080, etc
- **NOTA:** os portos TCP são independentes dos portos UDP (usados na comunicação com mensagens)...

Material de suporte às aulas de Redes de Computadores – Copyright DI – FCT/ UNL / 7

Exemplo(ECHO)

- O cliente liga-se ao servidor, identificado por um nome de máquina/endereço IP e um porto...
- A informação lida da consola é enviada pelo cliente ao servidor, que devolve uma sequência idêntica ao **copiar** os dados lidos do **canal de entrada** da ligação para o **canal de saída** da mesma.

Material de suporte às aulas de Redes de Computadores – Copyright DI – FCT/ UNL / 8

Código Java (ECHO)

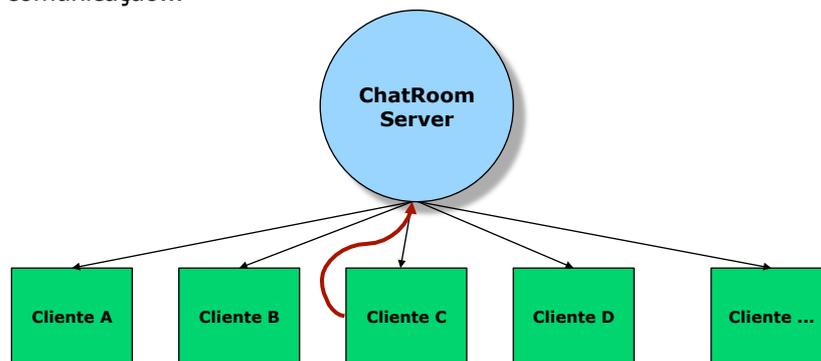
- Servidor ECHO ([EchoServer.java](#))
- Cliente ECHO ([EchoClient.java](#))

Programação em Java com Canais de dados TCP

- Pacotes
 - java.net
 - java.io
- Classes específicas
 - java.net.**ServerSocket** → “socket” do **servidor** para aceitar conexões
 - java.net.**Socket** → “socket” usado pelo **cliente** para estabelecer a conexão TCP e, posteriormente, por **este e pelo servidor** para enviar e receber dados.
 - java.io.**InputStream/OutputStream** → canais de entrada/saída associados a cada ligação TCP, que suportam, efectivamente, as operação de leitura e escrita sobre o canal TCP.

Exercício: "ChatRoom / Talk a N"

Com base no código do exemplo que lhe foi dado, implemente o seguinte sistema de ChatRoom usando canais de dados para suportar toda a comunicação...



Material de suporte às aulas de Redes de Computadores – Copyright DI – FCT/ UNL / 11

Exercício: "ChatRoom"

- Modificar o cliente dado de modo a introduzir concorrência, por recurso a um *thread* auxiliar.
 - O mesmo canal de dados ligação deverá ser usado para enviar e receber os dados e deverá ser usado enquanto o utilizador permanecer no ChatRoom.
- No servidor, o problema das leituras bloqueantes concorrentes é ainda mais agudo e complexo...

Novamente, o recurso a threads auxiliares pode ser usado para resolver o problema... Por exemplo, pondo o thread principal a aceitar as novas ligações e a criar um thread auxiliar, dedicado para atender o respectivo utilizador...

Material de suporte às aulas de Redes de Computadores – Copyright DI – FCT/ UNL / 12