

# Sistemas Distribuídos

## Aula IV – Comunicação II

---

PROF. DR. RAFAEL TEIXEIRA SOUSA

# Remote Method Invocation (RMI)

---

Objetos distribuídos

Similar ao RPC, mas com instanciação dinâmica de novos objetos e suas interfaces

Suportado por CORBA e JAVA

Componentes: referência para objetos, interfaces, métodos, exceções e coleta de lixo

# RMI

---

RMI permite que o código que define o comportamento e o código que o implementa executem em máquinas distintas

- Em um sistema distribuído, clientes se preocupam com a definição do serviço e os servidores focam em como prover o serviço

Exemplo: A definição de um serviço remoto é codificado usando uma interface Java enquanto a implementação do serviço é codificado em uma classe Java

# Referência para o objeto

---

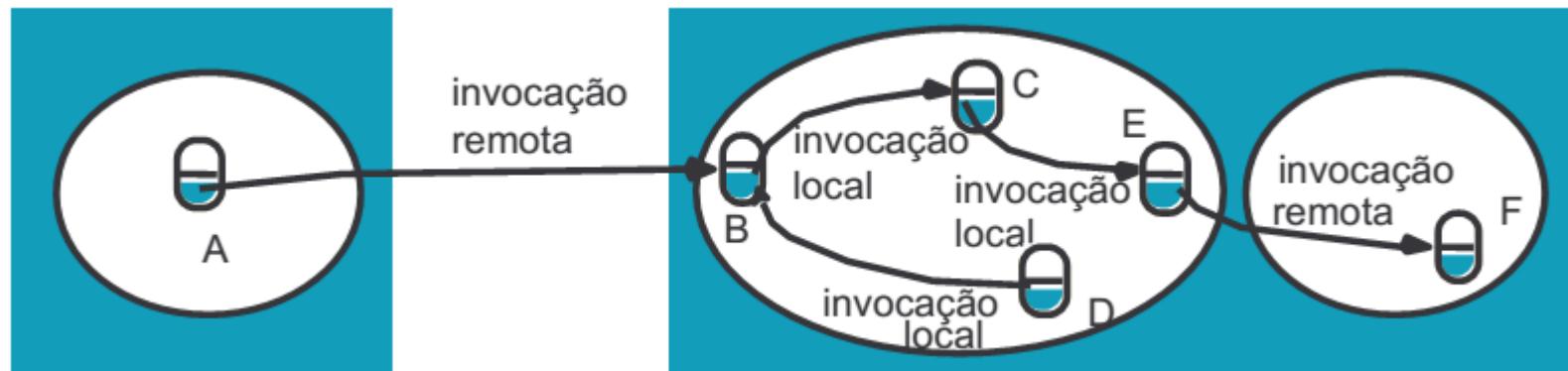
Antes de invocar um método em um objeto remoto é necessário uma referência para o objeto

Formas de encontrar essa informação:

- Protocolos de descoberta
- Serviços de nomeação como RMI Registry
  - Servidores registram objetos remotos com nomes
  - Clientes procuram por referências de objetos que casem com um nome
  - Nomes usam o formato URL
  - Rmi://maq:<port>/<NomeServiço>
  - Rmi://146.164.69.2:3128/Chat

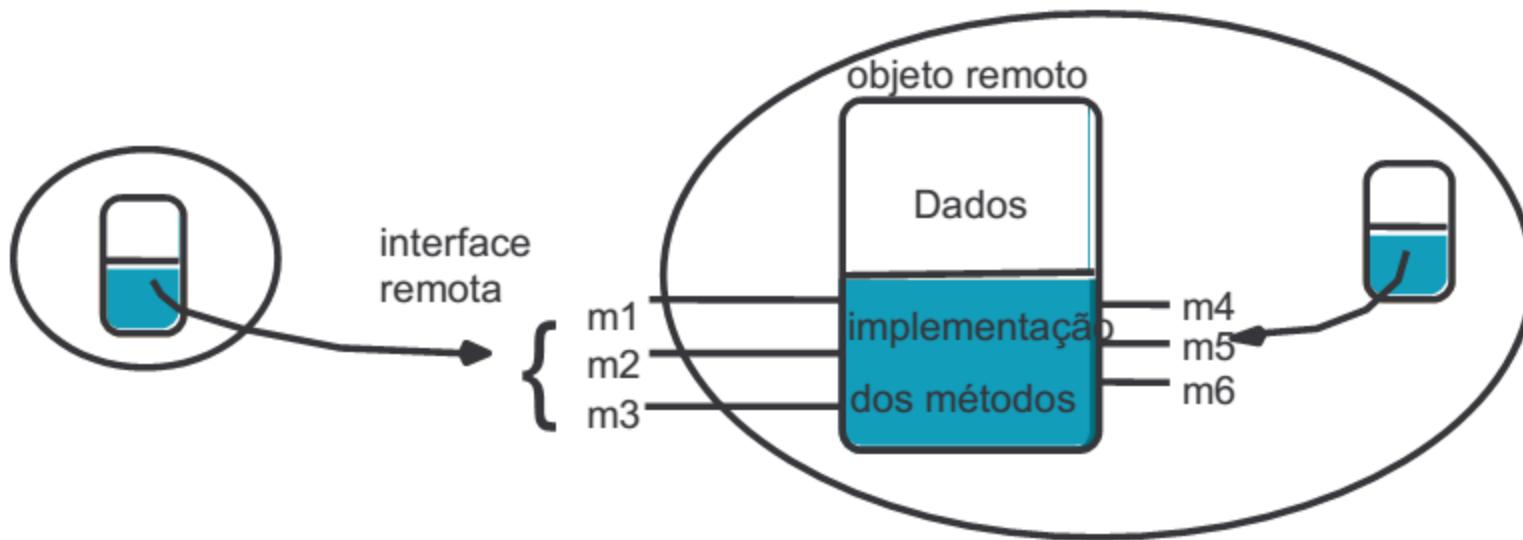
# Invocações Remotas e Locais

---



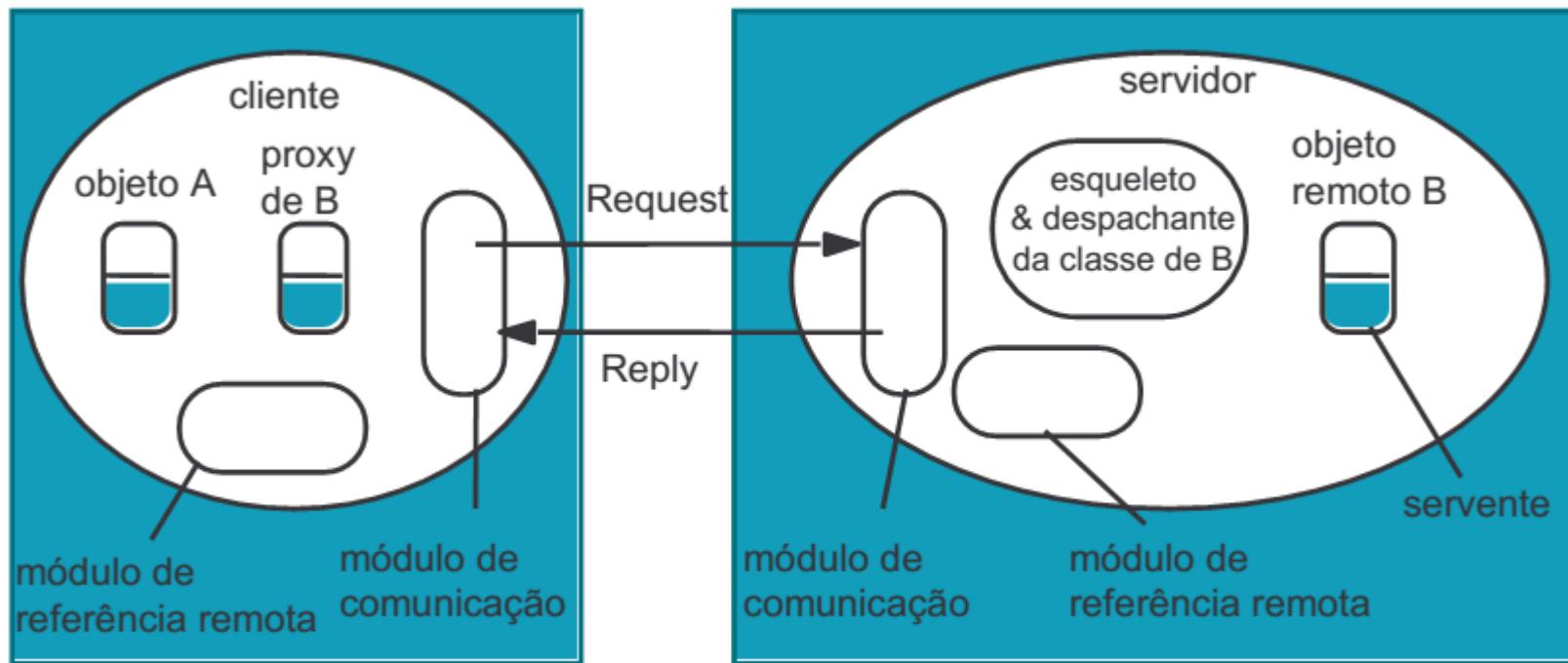
# Objeto remoto e sua interface

---



# RMI: proxy e esqueleto (skeleton)

---



# Implementação de RMI

---

## Módulo de comunicação

- Mensagens request/reply

## Módulo de referência remota

- Translação entre referências locais e remotas usando uma tabela de objetos remotos
- Usado quando (de)codificando referências a objetos remotos.

# Componentes do RMI

---

## No cliente

- Proxies: cópias locais de objetos remotos

## No servidor:

- Despachantes: recebe pedido e usa methodID para selecionar mensagens apropriadas no esqueleto
- Esqueleto: implementa métodos a interface remota
  - Decodifica os argumentos
  - Invoca o método correspondente no objeto remoto
  - Espera terminar a execução
  - Codifica os resultados, possíveis exceções e retorna-os em uma mensagem de reply ao método invocado pelo proxy

## Implementação

- Em alguns sistemas eles podem ser automaticamente compilados:
  - CORBA: usa descrições IDL de interfaces remotas para compilar a camada RMI gerando código C++
  - Java RMI: classes de proxy, esqueleto e despachante são geradas a partir da classe do objeto remoto