

# Implementação de um Serviço utilizando Comunicação Inter-processos através de Remote Method Invocation - RMI

Prof. Adriano Fiorese

## 1 Caracterização do Serviço

O serviço a ser desenvolvido é o de echo remoto com replicação e tolerância a falhas. Nesse trabalho, o(s) cliente(s) deve(m) executar as seguintes operações:

- Echo: Permite a invocação do método echo no servidor de echo. Tal operação passa como argumento a mensagem a ser devolvida como um echo, e que também será replicada no(s) servidor(es) clone(s) replica. Na Figura 1 essa operação está representada como envio e retorno da msg (operação echo).
- Obter a Lista de Msg já ecoadas: Essa operação permite invocar o método getListOfMsg no(s) servidor(es) de echo de forma a obter a lista completa de mensagens já enviadas anteriormente. Assim, quando o cliente invocar esse método no servidor, o retorno do método será a lista de mensagens que representa o histórico de mensagens já enviadas pelo(s) cliente(s).

O sistema funciona de forma replicada, como forma de garantir tolerância à falhas. Isso significa que as operações realizadas por uma entidade (servidor) devem ser replicadas em todas as demais réplicas (servidores). Ou seja, além das operações que podem ser executadas pelos clientes, o servidor também deverá executar (invocar) uma operação de replicação no(s) outro(s) servidore(s) réplicas. Na Figura 1 essa operação é ilustrada como replica msg.

Portanto, nesse caso, será possível que quando o cliente invoque o método echo do servidor mestre, este servidor durante a execução desse método invoque o método replica (passando a msg como argumento) no(s) servidor(es) clone(s). Tanto o servidor mestre quanto o servidor replica deverão manter todas as mensagens recebidas através da invocação do método echo em uma lista de mensagens que poderá ser acessada quando da invocação do método getListOfMsg por parte do cliente.

Importante notar que quando o servidor mestre não estiver ativo (for derrubado, desligado, etc...), a invocação dos métodos echo, e getListOfMsg deverá ser feita transparentemente para o próximo servidor réplica (clone), de forma que a falha de omissão do servidor mestre seja tolerada, de forma transparente para o usuário. Para tal será possível que ao iniciar os servidores seja informada a quantidade de servidores réplica existentes no sistema. Naturalmente que para tal funcionar de acordo, os servidores devem ser instanciados antes que o(s) cliente(s).

Cada equipe deverá elaborar um trabalho que deverá ter a implementação funcional, bem como um artigo de no mínimo 3 páginas no formato IEEE. **Nesse artigo deve constar na seção de introdução a motivação a respeito do problema da tolerância à falhas em sistemas distribuídos; na seção trabalhos relacionados deve constar uma pesquisa bibliográfica a respeito desses problemas e sistemas que os solucionam ou que apresentam solução parcial; na seção desenvolvimento deve constar a especificação (podem usar a figura deste texto inclusive, linguagem utilizada, bibliotecas, módulos, protocolo de aplicação desenvolvido, etc.) da solução que vcs desenvolveram para tratar desses problemas, através desse use case que é o echo tolerante a falha com replicação, como prova de conceito.**

## 2 Implementação

A implementação de tal serviço deverá ser realizada utilizando-se comunicação Inter-Processos por meio de RMI. Aconselho a utilizar o middleware RMI Java. O modelo arquitetural a ser utilizado deverá ser o cliente/servidor, na variação múltiplos servidores.

Na Figura 1, podemos observar as invocações de métodos necessárias para a perfeita execução do serviço solicitado.

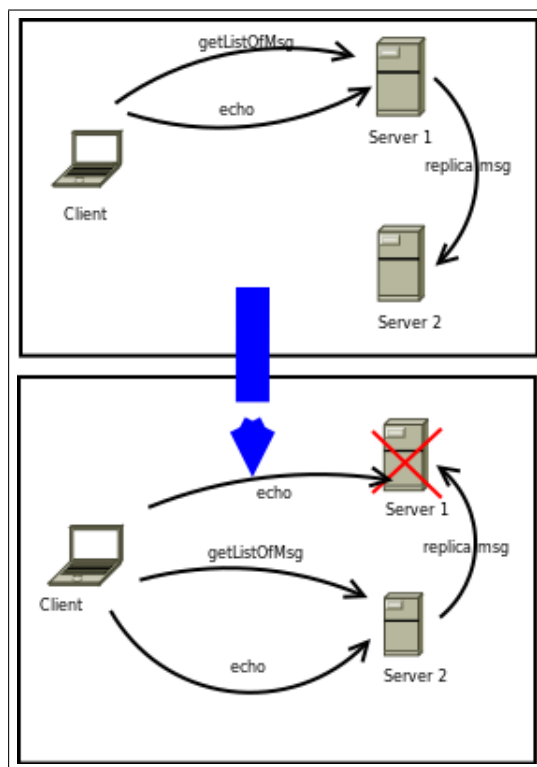


Figura 1: Serviço de Echo de Mensagens com replicação tolerante a falhas

## 3 Prazos

A entrega ocorrerá via moodle, envolvendo executáveis e código fonte bem como, artigo fonte (.doc ou .odt ou .tex) e pdf no dia 20/06/2016.

## 4 Equipes

Equipes de tamanho máximo igual a 2.

Tabela 1: Equipes

<b>Equipe</b>	<b>Nomes</b>
I	
II	
III	
IV	
V	
VI	