

Sistemas Distribuídos - Relatório Final EP1

Fernanda Sayuri Alves Ito – RA: 11201811042

Link do vídeo: https://www.youtube.com/watch?v=FI7L-f_dwic

Funcionalidades do Peer:

Os peers são criados, na função `Inicializa()` (linha 152), a partir da inserção do ip, porta do peer principal e de mais dois Peers vizinhos com seus respectivos ips e portas, no caso, como foi utilizado para a testagem o mesmo computador mas com quatro terminais diferentes foi utilizado o mesmo ip para todos os peers, o principal e seus adjacentes.

A partir disso, depois de instanciar todos os peers desejados, que podem ser mais de quatro, na função `Monitoramento()` (linha 240) é mostrado quais arquivos estão no caminho de pasta passado de 30 em 30 segundos, dessa maneira, caso ocorra a inserção ou a deleção de algum arquivo a qualquer tempo é printado no console essa alteração.

Além disso, a principal funcionalidade do peer é de realizar busca em seus peers vizinhos, através da função `sendMessage()` (linha 134), inserindo o nome do arquivo juntamente do seu tipo, dessa maneira, é enviado ao servidor UDP (linha 55) essa chamada através de um objeto mensagem, que possui todas as características do peer que fez a solicitação da busca e do arquivo procurado. Também foi usada duas funções: a `convertFromBytes()` (linha 38) e a `convertToBytes()` (linha 30) para codificar a mensagem e decodificá-la ao ser enviada e recebida.

Ou seja, um peer faz uma busca e essa mensagem é enviada a um peer vizinho de forma aleatória, quando esse peer vizinho recebe tal informação é feito a verificação se ele possui o arquivo procurado (linha 93); caso ele possua, é retornado ao peer principal o ip e porta do peer que possui o arquivo por meio de uma nova mensagem, mas com o tipo `response` ao invés de `search` (linha 95); caso ele não possua, é realizado novamente o envio de uma mensagem para outro peer vizinho deste até que seja encontrado o arquivo (linha 111). Contudo, caso sejam realizadas x requisições, linha 103, (no código foi estabelecido 10

tentativas de busca) entre os vizinhos dos peers e o arquivo não seja encontrado, a solicitação é declarada como “arquivo não encontrado”.

Uso das threads:

Foi utilizada uma thread para realizar o monitoramento que printa os arquivos de tempos em tempos de cada determinada pasta de cada Peer. Uma outra thread para a criação dos Peers e uma outra thread para a criação do Servidor UDP, que recebe as requisições search do peer cliente e envia o response.

Links dos lugares de onde baseou seu código:

<https://stackoverflow.com/questions/3736058/java-object-to-byte-and-byte-to-object-converter-for-tokyo-cabinet>

<https://www.baeldung.com/object-to-byte-array>