

INF01151 – SISTEMAS OPERACIONAIS II N
SEMESTRE 2018/1
TRABALHO PRÁTICO PARTE 2: REPLICAÇÃO E ELEIÇÃO DE LÍDER

ESPECIFICAÇÃO DO TRABALHO

Este projeto consiste na implementação de um serviço semelhante ao Dropbox e está dividido em duas partes. Na primeira etapa, foi necessário implementar o funcionamento básico do serviço, com enfoque nos aspectos de programação com múltiplos processos/threads, comunicação e controle de concorrência. Nessa etapa, você deverá estender o serviço com algumas funcionalidades avançadas, onde destaca-se: **esquema de replicação com eleição de líder**.

O programa deverá executar obrigatoriamente em ambientes Unix (Linux) mesmo que o trabalho tenha sido desenvolvido em outra plataforma, e implementado na linguagem C/C++. A especificação do relatório final a ser entregue é descrita ao final desse documento.

FUNCIONALIDADE DE REPLICAÇÃO E ELEIÇÃO DE LÍDER

Na segunda parte deste projeto, a aplicação deverá ser estendida para atender os conceitos estudados na segunda metade da disciplina. Isto inclui: esquema de replicação com eleição de líder.

1. Eleição de Líder

Algoritmos de eleição de líder permitem escolher, dentro de um conjunto de processos distribuídos, qual processo deve desempenhar um papel particular (e.g., coordenador, alocador de recursos, verificador, etc). Algoritmos de eleição de líder são muito usados como parte de outros algoritmos distribuídos, que exigem a escolha de um processo para desempenhar um papel específico.

Na primeira parte do trabalho, assumiu-se a existência de um único servidor Dropbox. Agora, no entanto, assumiremos que diferentes processos potencialmente poderão assumir o papel de servidor primário, mas o processo escolhido em questão deverá ser selecionado através de um dos algoritmos de eleição de líder vistos em aula: algoritmo do anel ou algoritmo do valentão (aula 21, e Capítulo 15 do livro do Coulouris).

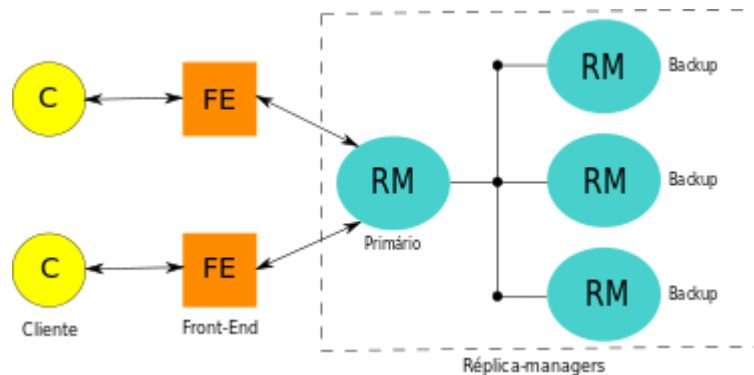
O algoritmo escolhido deverá ser implementado utilizando um mecanismo de troca de mensagens, podendo esse ser o próprio socket UDP, já utilizado no trabalho, ou algum outro mecanismo.

2. Replicação Passiva

O servidor implementado na Parte I, caso por ventura falhar, acarretará em indisponibilidade de serviço para seus clientes. Este é um estado que não é desejável no sistema. Para aumentar a disponibilidade do sistema, você deverá implementar um esquema de replicação passiva (aula 27, e Capítulo 18 do livro do Coulouris). Utilizando esse esquema de replicação, o servidor Dropbox será representado por uma instância de replica manager (RM) primária (selecionado via o algoritmo de eleição de líder), e uma ou mais instâncias de replica managers secundárias (ou backup).

Podemos entender este modelo adicionando um *front-end* (FE) entre a comunicação do cliente (C) e servidor, agora representado por um conjunto de *replica managers* (RMs). Este *front-end* será responsável por realizar a comunicação entre estas entidades, tornando transparente para o cliente qual é a cópia primária do servidor. Você precisará garantir que:

- (1) todos os clientes sempre utilizarão a mesma cópia primária;
- (2) após cada operação, o RM primário irá propagar o estado dos arquivos aos RMs de backup;
- (3) somente após os backups serem atualizados o primário confirmará a operação ao cliente.



DESCRIÇÃO DO RELATÓRIO A SER ENTREGUE

Deverá ser produzido um relatório fornecendo os seguintes dados:

- Descrição do ambiente de teste: versão do sistema operacional e distribuição, configuração da máquina (processador(es) e memória) e compiladores utilizados (versões).
- Apresente claramente no relatório uma descrição dos pontos abaixo:
 - o (A) Explique o funcionamento do algoritmo de eleição de líder implementado e justifique a sua escolha;
 - o (B) Como a replicação passiva foi implementada na sua aplicação e quais foram os desafios encontrados;
- Também inclua no relatório problemas que você encontrou durante a implementação e como estes foram resolvidos (ou não).

A **nota será atribuída baseando-se nos seguintes critérios**: (1) qualidade do relatório produzido conforme os itens acima, (2) correta implementação das funcionalidades requisitadas e (3) qualidade do programa em si (incluindo uma interface limpa e amigável, documentação do código, funcionalidades adicionais implementadas, etc).

DATAS E MÉTODO DE AVALIAÇÃO

O trabalho deve ser feito em grupos de **3 OU 4 INTEGRANTES**, conforme a configuração de grupos da Parte I. Não esquecer de identificar claramente os componentes do grupo no relatório.

Faz parte do pacote de entrega os arquivos fonte e o relatório em um arquivo ZIP. O trabalho deverá ser entregue até às **08:30 do dia 21 de junho (turma A)** ou às **08:30 do dia 18 de junho (turma B)**. A entrega deverá ser via moodle (link para submissão na Aula 29). As demonstrações ocorrerão no mesmo dia, no horário da aula.

Após a data de entrega, o trabalho deverá ser entregue via e-mail para alberto@inf.ufrgs.br (subject do e-mail deve ser "INF01151: Trabalho Parte 2"). Neste caso, será descontado 02 (dois) pontos por semana de atraso. O atraso máximo permitido é de duas semanas após a data prevista para entrega. Isto é, nenhum trabalho será aceito após o dia 05 de julho (turma A) ou dia 02 de julho (turma B).