

INF01151 – SISTEMAS OPERACIONAIS II N  
SEMESTRE 2025/1  
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, focando 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**.

A aplicação deve obrigatoriamente executar em **ambientes Unix (Linux)**, mesmo que tenha sido desenvolvida em outras plataformas. O programa deverá ser implementado utilizando a API **Transmission Control Protocol (TCP) sockets** do Unix e utilizando **C/C++**.

## FUNCIONALIDADES AVANÇADAS

---

### 1. Replicação Passiva

O servidor implementado na Parte I, caso venha a falhar, levará à indisponibilidade de serviço para seus clientes. Este é um estado que não é desejável no sistema. Para aumentar a disponibilidade do sistema na ocorrência de falhas do servidor principal, um novo servidor deverá assumir o seu papel e manter o serviço de gerenciamento de arquivos funcionando. Note que essa mudança deve ser transparente para os usuários e seus arquivos devem manter-se disponíveis, mesmo após a falha. Para garantir que as modificações de arquivos estarão disponíveis em um novo servidor, você deverá utilizar um esquema de **replicação**, informando todas as modificações realizadas aos servidores secundários.

Mais especificamente, você deverá implementar um esquema de **Replicação Passiva**<sup>1</sup>, onde o servidor é representado por uma instância primária de réplica manager (RM), e uma ou mais instâncias de RM secundárias (ou backup). Podemos entender este modelo adicionando um *front-end* (FE) entre a comunicação do cliente (C) e o conjunto de RMs. O *front-end* é responsável por realizar a comunicação entre os clientes e o serviço de replicação, tornando transparente para o cliente qual é a cópia primária do servidor (Figura 1).

Seu esquema 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.

---

<sup>1</sup> (Tópico 26 no moodle, e Capítulo 18 do livro COULOURIS, G.; DOLLIMORE, J.; KINDBERG, T.; BLAIR, G. – "Distributed Systems: Concepts and Design" (5th edition), Addison-Wesley, 2012).

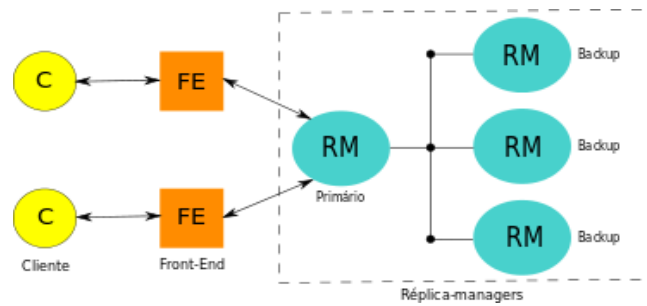


Figura 1 - Esquema de replicação passiva

## 2. 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, como diferentes processos potencialmente poderão assumir o papel de servidor primário, o processo escolhido em questão deverá ser selecionado através de um dos algoritmos de eleição de líder vistos em aula<sup>2</sup>: algoritmo do anel ou algoritmo do valentão.

Quando o servidor principal falhar, o algoritmo de eleição de líder deverá ser utilizado para determinar o próximo servidor primário. Nesse caso, um dos servidores backup deverá assumir essa função, mantendo um estado consistente do sistema. Para isso, implemente um dos algoritmos vistos em aula para eleger um novo RM primário após uma falha. Lembre-se de garantir que seu mecanismo atualize as informações sobre o novo líder nos FE dos clientes.

## DESCRIÇÃO DO RELATÓRIO

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).

<sup>2</sup> Algoritmo do anel ou algoritmo do valentão (Tópico 20 no moodle, e Capítulo 15 do livro COULOURIS, G.; DOLLIMORE, J.; KINDBERG, T.; BLAIR, G. – "Distributed Systems: Concepts and Design" (5th edition), Addison-Wesley, 2012).

## 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 23 de junho**. A entrega deverá ser via moodle. As demonstrações ocorrerão no mesmo dia, no horário da aula.

Após a data de entrega, o trabalho poderá ser entregue via e-mail para [alberto@inf.ufrgs.br](mailto: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 **uma semana** após a data prevista para entrega. Isto é, todos os trabalhos deverão ser entregues e apresentados até o dia 30 de junho.