

# MAC0499 - Trabalho de Formatura Supervisionado

## Proposta de monografia

---

**Alunos:** Danielle Tiemy Paulo Miazaki  
Renato Urquiza Lundberg  
**Curso:** Bacharelado em Ciência da Computação  
**Orientador:** Alfredo Goldman vel Lejbman  
**Tipo de trabalho:** Projeto  
**Título:** MOOFS – Mobile Objects Oriented File System

---

## 1 Tema

### 1.1 Sistema de arquivos

Um sistema de arquivos é um método de armazenar e organizar dados em um espaço de armazenagem. Para isso, o sistema de arquivos pode dividir os dados em blocos menores de informação.

#### 1.1.1 Sistema de arquivos distribuído

(Esta seção é baseada no *Guia de Inovação Tecnológica em Cluster e Grid*<sup>1</sup>.)

Um sistema de arquivos distribuído (SAD) visa fornecer acesso remoto a dados de forma transparente ao usuário.

Num SAD, os arquivos podem estar centralizados em apenas um servidor ou espalhados em várias máquinas. Além disso, para aumentar a tolerância a falhas, podem-se ter várias réplicas de um arquivo inteiro ou de partes dele. Como em praticamente qualquer sistema de arquivos, mais de um usuário pode acessar o mesmo recurso ao mesmo tempo.

Porém, o SAD deve sempre dar aos seus usuários a impressão de que estão manipulando um arquivo local com apenas uma cópia, tendo acesso exclusivo.

---

<sup>1</sup><http://guialivre.governoeletronico.gov.br/guiaonline/guiacluster/node45.php>

## 2 Objetivos

Este trabalho visa o desenvolvimento do *Mobile Objects Oriented File System (MOOFS)*, um sistema de arquivos distribuído em que cada arquivo encontra-se distribuído entre vários computadores. Nenhuma máquina armazena o arquivo inteiro, apenas pequenos blocos de dados.

No *MOOFS*, o bloco é representado por um conjunto de objetos distribuídos na rede. Existe comunicação entre esse conjunto de objetos, que é utilizada para detectar falhas, como um computador sobrecarregado ou mesmo uma total perda de comunicação. Como resposta a esses eventos, objetos são capazes de se movimentar entre computadores e criar cópias de si mesmos (gêmeos) para aumentar o número de objetos guardando o bloco, garantindo que a informação que eles guardam não será perdida. O sistema é tolerante a falhas e descentralizado.

Os blocos não sabem o que vem antes ou depois. O dono do sistema é quem possui a informação de como montar os blocos em um arquivo funcional, e pode acessar os dados do arquivo. Isso aumenta a segurança da informação, pois dificulta que alguém mal intencionado consiga recuperar algum dado, mesmo que obtenha acesso a todos os blocos.

A idéia central deste projeto é que o bloco não precisa estar em um único disco. Desse modo, o *MOOFS* possibilitaria o armazenamento de grandes quantidades de dados, a partir de qualquer computador. O potencial de sua utilização é muito alto em dispositivos móveis, que sofrem com a falta de espaço para armazenamento.

O *MOOFS* deve ser completamente transparente ao usuário e às aplicações, não sendo necessário nenhum tipo de interface. O sistema pode ser acessado por qualquer aplicação que necessite de armazenamento sem qualquer adaptação.

Partições com o *MOOFS* teriam um tamanho variável e virtualmente ilimitado, dependendo apenas do número de computadores que hospedam os blocos, e a velocidade de acesso é limitada apenas pela capacidade de download da conexão do usuário.

## 3 Ferramentas

A biblioteca *FUSE* permite a criação de sistemas de arquivos em ambientes *POSIX* (tais como *Linux*, *Unix* e *Mac*) com muita facilidade, sem que seja necessário lidar diretamente com chamadas do sistema operacional.

As chamadas a objetos remotos torna muito mais simples a comunicação, e, associadas às threads, torna o sistema bastante poderoso. Essas características levaram à escolha da linguagem *Python*, por ser orientada a objetos, ter um binding estável com a biblioteca *FUSE*, threads e chamadas a objetos remotos.

## 4 Atividades já realizadas

Até o momento nos empenhamos em conhecer melhor a linguagem *Python* e a biblioteca *FUSE*. Além disso, já foram desenvolvidos os primeiros protótipos do *MOOFS*, ainda na Fase 1 de implementação (vide seção 5, “Cronograma de atividades para o segundo semestre”).

## 5 Cronograma de atividades para o segundo semestre

Atividade	Mês				
	Julho	Agosto	Setembro	Outubro	Novembro
Fase 1	● ● ● ● ● ● ● ●				
Fase 2	● ● ● ● ● ● ● ●	● ● ● ● ● ● ● ●			
Fase 3		● ● ● ● ● ● ● ●	● ● ● ● ● ● ● ●	● ● ● ● ● ● ● ●	
Fase 4				● ● ● ● ● ● ● ●	● ● ● ● ● ● ● ●

[Fase 1] Implementação: blocos locais e persistidos no disco.

[Fase 2] Implementação: blocos remotos e únicos, persistidos no disco.

[Fase 3] Implementação: blocos remotos e com gêmeos, persistidos no disco.

[Fase 4] Elaboração do pôster, da apresentação e da monografia.

## 6 Estrutura esperada da monografia

### • Parte Técnica

1. **Introdução:** descrição das motivações do projeto.
2. **Objetivo:** explicação do objetivo do projeto e que funcionalidades deve oferecer.
3. **Conceitos e tecnologias associados:** explicação de conceitos importantes e ferramentas utilizadas no desenvolvimento do projeto, além de descrição de trabalhos associados.
4. **Atividades realizadas:** descrição do andamento de cada fase da implementação e conseqüentes decisões de projeto.
5. **Descrição técnica:** detalhes do funcionamento interno do projeto.
6. **Resultados e produtos obtidos:** descrição do produto final do projeto.

7. **Conclusões:** enumeração de conclusões pós-projeto e de possibilidades para o futuro do mesmo.

8. **Bibliografia**

- **Parte Subjetiva:** comentários de cada integrante do grupo sobre a participação no projeto.