

## **Computação Distribuída**

### **O que é computação distribuída**

A computação distribuída é uma abordagem de sistemas computacionais onde vários computadores independentes trabalham de forma colaborativa e coordenada para alcançar um objetivo comum. Esses sistemas são projetados para compartilhar recursos como dados, poder de processamento e armazenamento, interligados por uma rede de comunicação. Uma de suas principais características é a transparência, proporcionando aos usuários uma experiência unificada, independentemente de onde os recursos estejam localizados.

Segundo Tanenbaum e Van Steen (2013), a computação distribuída é fundamentada na ideia de que "um sistema distribuído é uma coleção de computadores independentes que parecem aos usuários como um sistema único e coeso".

### **Conceitos de escalabilidade e tolerância a falhas**

**Escalabilidade:** Refere-se à capacidade do sistema de acomodar o aumento de carga de trabalho ou de recursos adicionados. Essa escalabilidade pode ser horizontal, com a adição de novos servidores, ou vertical, aumentando os recursos de máquinas existentes.

**Tolerância a falhas:** Garante que o sistema continue a operar, mesmo que uma ou mais partes falhem. Isso é obtido por meio de técnicas como replicação de dados e detecção de falhas, que redirecionam tarefas automaticamente para componentes funcionais.

Esses conceitos são essenciais para projetos de sistemas distribuídos, assegurando desempenho e confiabilidade.

### **Vantagens e desvantagens**

#### **Vantagens:**

- Melhor utilização de recursos.
- Alta disponibilidade.
- Flexibilidade para expandir.

#### **Desvantagens:**

- Maior complexidade no desenvolvimento e gerenciamento.

Possibilidade de inconsistências de dados.  
Latência devido à comunicação entre máquinas.

## Arquitetura da Solução

### Diagrama de comunicação

O sistema é composto por:

**Cliente:** Envia requisições de busca.

**Servidor A:** Recebe as requisições e as distribui para outros servidores.

**Servidor B:** Processa partes da busca e retorna os resultados ao Servidor A.

### Formato do dado

Os dados trafegados entre os componentes foram definidos como JSON devido à sua leveza e fácil manipulação. Um exemplo seria:

**Servidor A → Servidor B:** {"query": "produto", "pagina": 1}.

**Servidor A → Cliente:** {"resultados": [{"id": 1, "nome": "Produto A"}]}.

### Algoritmo de busca

Para este projeto, foi escolhido o algoritmo **MapReduce**, que divide a tarefa de busca em partes menores processadas paralelamente pelos servidores. Isso reduz o tempo de execução e facilita a escalabilidade. O MapReduce é adequado devido à sua eficiência em grandes volumes de dados, permitindo operações distribuídas de maneira organizada e robusta.

### Referências

Tanenbaum, A. S., & Van Steen, M. *Distributed Systems: Principles and Paradigms*. Pearson, 2013.

Artigos obtidos em IEEE Xplore: "**Scalability in Distributed Systems**".

Participantes do grupo:

João Victor Cortez

Cauê Justen Garbi

