
EXPLORAÇÃO DOS CONCEITOS FUNDAMENTAIS E ARQUITETURAS EM SISTEMAS DISTRIBUÍDOS

Autor RIAN DA SILVA FERREIRA

Instituição ANHANGUERA

1. Introdução

A área da robótica, como a construção de robôs seguidores de linha utilizando microcontroladores Arduino, é um exemplo da crescente complexidade e interconexão de sistemas tecnológicos. Neste contexto, os **sistemas distribuídos** emergiram como uma arquitetura fundamental e de grande interesse, com aplicações em diversas áreas da sociedade. Um sistema distribuído é caracterizado como um conjunto de computadores interligados por uma rede que, para o utilizador final, aparenta ser um sistema único e coerente, mesmo que os componentes operem de forma independente e frequentemente com hardware e software distintos.

A relevância dos sistemas distribuídos é inegável em aplicações modernas, abrangendo desde jogos *multiplayer online*, como o Smite, até sistemas de transações financeiras, motores de busca e as redes sociais, incluindo a própria Internet. Para profissionais de TI, compreender a fundo estes sistemas – seus componentes, objetivos e desafios – é crucial para o desenvolvimento de aplicações robustas e resilientes a falhas. Este trabalho tem como objetivo explorar os **conceitos fundamentais, modelos arquitetónicos, objetivos, desafios e tecnologias de suporte** que tornam os sistemas distribuídos eficazes, proporcionando uma base sólida para a compreensão e o avanço no campo da tecnologia da informação.

2. Objetivo

O objetivo deste trabalho é **explorar a utilização e a fundamentação teórica dos sistemas distribuídos**, analisando os conceitos essenciais e os modelos de arquitetura específicos que permitem a interligação de múltiplos computadores para operarem como um sistema único e coeso para o utilizador final. Adicionalmente, visa-se **demonstrar a aplicação prática de microcontroladores combinados com outros dispositivos de entrada** para executar tarefas específicas em ambientes distribuídos, através da discussão dos objetivos intrínsecos a estes sistemas, os desafios inerentes à sua implementação, e as tecnologias de suporte que os tornam viáveis.

3. Material e Métodos

Para alcançar o objetivo proposto, este trabalho baseia-se na **análise e síntese de informações sobre os modelos de arquitetura, conceitos fundamentais, objetivos, desafios, integração, comunicação e tecnologias de suporte** que sustentam os sistemas distribuídos, conforme detalhado nas fontes primárias da área.

Modelos de Arquitetura de Redes de Computadores: Existem três modelos arquitetônicos principais que servem de base para sistemas distribuídos:

- A **Arquitetura Cliente-Servidor**, a mais conhecida e utilizada desde a década de 70, envolve servidores que disponibilizam serviços (ex: bases de dados, autenticação) a máquinas cliente (laptops, smartphones).
- A **Arquitetura Ponto a Ponto (P2P)**, surgida nos anos 80, onde todos os computadores (nós) atuam tanto como dispositivos finais quanto como servidores, compartilhando recursos. Exemplos incluem partilha de ficheiros via Bluetooth ou BitTorrent.
- A **Arquitetura Descentralizada**, mais recente (anos 2000), é um modelo híbrido que replica o estado da aplicação entre os próprios computadores que atuam como servidores, garantindo consenso e transparência, como nas plataformas Blockchain.

Conceitos Fundamentais e Objetivos dos Sistemas Distribuídos: Os sistemas distribuídos são concebidos em torno de princípios-chave para superar desafios:

- **Compartilhamento de Recursos:** Permite acesso partilhado a recursos como bases de dados e serviços de autenticação, gerando economia financeira, mas aumentando a superfície de ataque para *hackers*.
- **Confiabilidade:** Capacidade do sistema de funcionar conforme planeado, sendo geralmente maior em sistemas distribuídos. Baseia-se no **Teorema CAP** (Consistência, Disponibilidade, Resiliência), que afirma que apenas dois desses pilares podem ser plenamente alcançados simultaneamente.
- **Desempenho:** Geralmente superior aos sistemas centralizados devido a múltiplas instâncias de hardware e software. Medido por tempo de resposta, *throughput* e consumo de recursos de rede.
- **Granularidade:** Refere-se à capacidade de um sistema ser executado por vários processadores, dividindo tarefas em paralelo.
- **Desafios (Aspectos de Projeto):** Os sistemas distribuídos enfrentam metas como **Abertura** (modularidade), **Concorrência** (acesso simultâneo), **Escalabilidade** (capacidade de expansão/redução, incluindo escalabilidade geográfica e administrativa), **Heterogeneidade** (funcionamento em diferentes ambientes de hardware/software), **Segurança** (proteção de dados), **Tolerância a Falhas/Resiliência** (capacidade de autorrecuperação, com grupos simples ou hierárquicos de processos), e **Transparência** (ocultação do funcionamento interno ao utilizador). O **Middleware** é um componente crucial para a heterogeneidade e a gestão da comunicação.

Integração e Comunicação: A comunicação entre máquinas é essencial.

- **Protocolos de Transporte:** **TCP** (orientado à conexão, garante retransmissão, para comunicações não sensíveis a atrasos) e **UDP** (não orientado à conexão, minimiza atraso, para tempo real).
- **Sockets:** Utilizam TCP ou UDP para comunicação interprocessos, combinando endereço IP e porta de comunicação. Possuem primitivas como *socket*, *bind*, *listen*, *accept*, *connect*, *send*, *receive*, *close* para gerir a comunicação.

- **RPC (Remote Procedure Call):** Uma forma mais granular de comunicação, permitindo que uma máquina cliente execute métodos implementados em outra máquina (servidor) como se fossem locais. **RMI (Remote Method Invocation)** é a implementação de RPC para linguagens orientadas a objetos como Java.

Tecnologias de Suporte: Fundamentais para a implementação eficiente:

- **Virtualização:** Permite criar versões virtuais de hardware, redes e armazenamento. Máquinas virtuais (VMs) são abstrações de hardware que permitem a coexistência de múltiplos sistemas operativos em uma única máquina física (hospedeiro), gerenciadas por *hypervisores* como VirtualBox e VMWare. Traz vantagens de redução de espaço, custos administrativos, energia e otimização de recursos.
- **Containerização:** Considerada mais leve e eficiente que a virtualização, cria ambientes virtuais isolados para aplicações. Contêineres empacotam a aplicação com todas as bibliotecas, compartilhando o *kernel* do SO hospedeiro, resultando em menor ocupação de recursos. O **Docker** é a implementação mais popular, e ferramentas de orquestração como **Kubernetes** e **Swarm** gerenciam múltiplos contêineres, especialmente em arquiteturas de **microsserviços**.

Tipos de Computação Distribuída:

- **Computação em Cluster:** Conjunto de máquinas homogêneas, ligadas por rede local, que dividem a carga de processamento para tarefas específicas de alto poder. Um nó mestre gerencia as tarefas.
- **Computação em Grid:** Conjunto de máquinas heterogêneas que interligam vários *clusters* para executar diferentes tarefas relacionadas, formando centros de pesquisa multidisciplinar.

4. Resultados e Discussão

A análise aprofundada dos sistemas distribuídos demonstra que a sua arquitetura permite superar as limitações dos sistemas centralizados, resultando em **maior confiabilidade, desempenho, escalabilidade e transparência**. A capacidade de **compartilhamento de recursos** nos sistemas distribuídos proporciona uma notável **economia financeira**, uma vez que reduz a necessidade de replicar recursos físicos em cada nó. Esta abordagem, no entanto, exige uma atenção acrescida à segurança, dado o aumento de pontos de acesso passíveis de exploração.

A **resiliência** inerente a estes sistemas, alcançada pela organização estratégica de processos em grupos e pela capacidade de auto recuperação, garante a **continuidade do serviço** mesmo perante falhas, o que é de suma importância para aplicações de missão crítica. A **heterogeneidade**, facilitada pela camada de *middleware*, é um dos maiores trunfos, permitindo a **interoperabilidade entre componentes de hardware e software distintos**, aumentando a flexibilidade e a adaptabilidade do sistema a ambientes diversos. A **escalabilidade**, tanto em termos geográficos para otimização de latência e resiliência a desastres, quanto administrativos para uma gestão eficiente, assegura que o sistema possa ajustar-se dinamicamente à demanda, otimizando o consumo de recursos e a experiência do utilizador.

A evolução das **tecnologias de virtualização e containerização** tem sido catalítica para a implementação eficiente e modular de sistemas distribuídos. A containerização, em particular com o **Docker** e a orquestração via **Kubernetes ou Swarm**, tem impulsionado a adoção de **arquiteturas de microsserviços**, promovendo um baixo acoplamento entre os componentes, facilitando a implantação e permitindo a escalabilidade independente de cada serviço. Este paradigma, ao mesmo tempo que apresenta desafios em termos de segurança e complexidade de gestão, consolida a importância dos sistemas distribuídos como a espinha dorsal das aplicações modernas, permitindo a criação de **soluções altamente disponíveis, performáticas e resilientes**.

5. Conclusão

Em suma, este trabalho demonstra que os **sistemas distribuídos representam uma plataforma robusta e escalável** para a construção de aplicações complexas, interligando e coordenando múltiplas máquinas de forma coesa. A exploração dos modelos arquitetônicos, como cliente-servidor, ponto a ponto e descentralizada, juntamente com os conceitos fundamentais de compartilhamento de recursos, confiabilidade (Teorema CAP), desempenho, escalabilidade e tolerância a falhas, realça a capacidade destes sistemas de se adaptarem e operarem de maneira eficiente em cenários diversos. A crucialidade do *middleware* na facilitação da comunicação e na gestão da heterogeneidade, bem como o papel transformador das tecnologias de virtualização e containerização, evidenciam a complexidade e a engenhosidade por trás da sua implementação. A aplicação do Arduino na robótica, por exemplo, ilustra como a combinação de sensores, motores e programação permite a navegação autônoma e tomada de decisões, refletindo a essência da automação e distribuição de tarefas em sistemas complexos. Os sistemas distribuídos não só otimizam a utilização de recursos, mas também promovem a resiliência e a transparência para o utilizador final, consolidando-se como a **base para o desenvolvimento de soluções inovadoras na era digital**.

6. Referências

- ANHANGUERA - CENTRO UNIVERSITÁRIO ANHANGUERA DE SÃO PAULO. **UTILIZAÇÃO DO MICROCONTROLADOR ARDUINO PARA A CONSTRUÇÃO DE UM ROBÔ SEGUIDOR DE LINHA**. (ARTIGO ROBÔ.pdf).
- Comissão Organizadora do 28º Encontro de Atividades Científicas. **Regulamento Geral**. (regulamento.pdf).
- COULOURIS, G. et al. **Sistemas Distribuídos - Conceitos e Projeto**. 5. ed. Porto Alegre: Editora Bookman, 2013.
- DIEDRICH, C. **Docker Swarm**. Disponível em: <https://www.mundodocker.com.br/docker--swarm/>. Acesso em: 5 dez. 2018.
- DOCKER, C. **Docker Containerization Unlocks the Potential for Dev and Ops**. Disponível em: <https://www.docker.com/why-docker>. Acesso em: 10 dez. 2018.
- GREINER, R. **CAP Theorem: Revisited**. 14 ago. 2014. Disponível em: <http://robertgreiner.com/2014/08/cap-theorem-revisited/>. Acesso em: 25 mar. 2018.
- GOODRICH, M. T.; TAMASSIA, R. **Introdução à Segurança de Computadores**. Porto Alegre: Bookman, 2012.
- LINUX CONTAINER. **Infrastructure for container projects**. Disponível em: <https://linuxcontainers.org>. Acesso em: 10 dez. 2018.

- MAIA, L. P. **Arquitetura de redes de computadores**. Rio de Janeiro: LTC, 2013.
- NTP. **Network Time Protocol project**. [S.l.], 2018. Disponível em: <http://www.ntp.org/>. Acesso em: 18 out. 2018.
- ORACLE. **Oracle VM VirtualBox®: User Manual**. 2018. Disponível em: <https://www.virtu-albox.org/manual/UserManual.html>. Acesso em: 02 nov. 2018.
- PEREIRA, Caique Silva. **Sistemas distribuídos**. Londrina: Editora e Distribuidora Educacional S.A., 2019.
- REDHAT. **O que é virtualização?** 2018. Disponível em: <https://www.redhat.com/pt-br/topics/virtualization/what-is-virtualization>. Acesso em: 27 nov. 2018.
- "Resumo Livro 19/08" (fonte de conteúdo principal para o artigo).
- TANENBAUM, A. S; STEEN, M. V. **Sistemas Distribuídos - Princípios e Paradigmas**. 2. ed. São Paulo: Pearson, 2008.