

SISTEMAS DISTRIBUÍDOS: CONCEITOS, OBJETIVOS E DESAFIOS

Autor: Miguel Expedito

Instituto: Anhanguera

RESUMO:

Este relatório apresenta uma análise do conceito de sistemas distribuídos, definindo-os como um conjunto de computadores independentes que, para o usuário, operam como um sistema único. Discute os três objetivos principais desses sistemas: o compartilhamento de recursos, que visa economia, mas eleva desafios de segurança; a confiabilidade, frequentemente contextualizada pelo Teorema CAP (Consistência, Disponibilidade e Tolerância a Partições), que implica trade-offs inerentes; e o desempenho, que se beneficia do uso paralelo de múltiplos recursos. Explora também conceitos importantes, como a granularidade das tarefas, e detalha os principais desafios do projeto, incluindo abertura, concorrência, escalabilidade, heterogeneidade, segurança, tolerância a falhas e transparência.

Palavras-chave: Sistemas distribuídos. Compartilhamento de recursos. Confiabilidade. Teorema CAP. Desempenho. Escalabilidade. Segurança.

1 INTRODUÇÃO:

Os sistemas distribuídos tornaram-se um pilar fundamental da tecnologia moderna, permeando diversas aplicações do cotidiano, desde jogos multiplayer online até sistemas financeiros complexos. Para um profissional de TI, compreender a essência desses sistemas é crucial, pois eles representam a base para a criação de soluções robustas, escaláveis e resilientes a falhas. Este relatório, elaborado com base no resumo de "Sistemas Distribuídos", busca elucidar os conceitos fundamentais, os objetivos que impulsionam sua adoção e os desafios inerentes à sua implementação e gerenciamento.

2 DEFINIÇÃO, OBJETIVOS E CONCEITOS DE SISTEMAS DISTRIBUÍDOS:

2.1 Definição

Um sistema distribuído é caracterizado como um conjunto de computadores que atuam de forma independente, mas que se interligam por meio de uma rede, proporcionando ao usuário final a percepção de um sistema coeso e único. Embora as máquinas componentes possam ter hardwares e sistemas operacionais distintos, elas funcionam de maneira harmoniosa, dissimulando sua individualidade.

2.2 Objetivos Principais

A concepção e implementação de sistemas distribuídos são guiadas por três objetivos principais, conforme delineado na literatura:

- **Compartilhamento de Recursos:** Este objetivo permite que recursos diversos, como bancos de dados, links de rede e serviços de autenticação, sejam acessados e utilizados por diferentes

máquinas na arquitetura. A principal vantagem reside na economia financeira, ao reduzir a necessidade de replicar fisicamente cada recurso em todos os nós do sistema. Contudo, o compartilhamento introduz desafios significativos em termos de segurança, pois expande os pontos de acesso que podem ser vulneráveis a ataques. Portais de notícias que agregam informações de múltiplos servidores exemplificam a aplicação desse conceito.

- **Confiabilidade:** A confiabilidade refere-se à capacidade do sistema de desempenhar suas funções conforme o planejado, consistentemente e de forma adequada. Sistemas distribuídos tendem a ser mais confiáveis do que os centralizados. Um conceito fundamental para a análise da confiabilidade é o Teorema CAP, que postula um trade-off entre Consistência (todos os usuários veem os mesmos dados simultaneamente), Disponibilidade (o sistema permanece operacional para requisições) e Tolerância a Partições (o sistema continua funcionando mesmo com falhas de comunicação entre seus nós). O teorema indica que, em um sistema distribuído, é inviável garantir os três atributos em sua totalidade, sendo necessário priorizar dois deles, enfraquecendo o terceiro.
- **Desempenho:** A busca por um desempenho superior é um dos maiores impulsionadores da adoção de sistemas distribuídos. Ao utilizar múltiplas instâncias de hardware e software em paralelo, esses sistemas geralmente superam os centralizados em termos de velocidade e capacidade de processamento. O desempenho pode ser avaliado por métricas como tempo de resposta do servidor, throughput (taxa de transferência de dados), consumo de recursos da rede e resultados de benchmarks.

2.3 Outros Conceitos Importantes: Granularidade

A granularidade diz respeito ao nível de divisão das tarefas de um sistema para que estas possam ser executadas paralelamente por múltiplos processadores. Para uma implementação eficaz, o software deve ser estruturado com métodos que permitam essa divisão. A granularidade é classificada em fina, para tarefas pequenas e frequentes, onde os benefícios da execução paralela superam os custos de coordenação, e grossa, para tarefas maiores e menos frequentes, que visam maximizar a ocupação dos processadores disponíveis.

3 DESAFIOS DOS SISTEMAS DISTRIBUÍDOS

Apesar das inúmeras vantagens, o projeto e a manutenção de sistemas distribuídos envolvem a superação de diversos desafios complexos. As metas a seguir são vistas mais como obstáculos a serem minimizados do que como características plenamente alcançáveis:

- **Abertura:** Refere-se à facilidade com que o sistema pode ser integrado com novas tecnologias e frameworks, ou ter seus componentes alterados sem comprometer a integridade do todo.
- **Concorrência:** Capacidade de o sistema suportar e gerenciar o acesso e a utilização simultânea por múltiplos usuários.
- **Escalabilidade:** Habilidade do sistema de expandir ou contrair seus recursos (seja hardware ou software) para lidar com variações na demanda, como o aumento ou a diminuição de acessos simultâneos.
- **Heterogeneidade:** Capacidade de o sistema funcionar em um ambiente composto por diferentes tipos de hardware (com variadas capacidades de memória, processamento) e

software (diferentes sistemas operacionais). O middleware desempenha um papel crucial na conciliação dessas diferenças.

- **Segurança:** A proteção dos dados contra violações de confidencialidade e integridade é um aspecto de projeto de suma importância, dado o maior número de pontos de acesso em sistemas distribuídos.
- **Tolerância a Falhas:** Conhecida também como resiliência, é a capacidade do sistema de se recuperar automaticamente de falhas, mantendo a operação mesmo diante de problemas em um ou mais de seus componentes.
- **Transparência:** Medida de quão bem a complexidade subjacente do sistema é ocultada do usuário. Quanto maior a transparência, menor a necessidade do usuário de compreender a implementação e o funcionamento interno do sistema. Os tipos de transparência incluem acesso, localização, migração, replicação, concorrência e falha.

4 CONCLUSÃO:

Os sistemas distribuídos são, inegavelmente, a espinha dorsal de muitas das inovações tecnológicas atuais, oferecendo vantagens notáveis em termos de compartilhamento de recursos, confiabilidade e desempenho. No entanto, sua complexidade exige uma abordagem metódica para enfrentar os desafios relacionados à abertura, concorrência, escalabilidade, heterogeneidade, segurança, tolerância a falhas e transparência. A compreensão aprofundada desses aspectos é essencial para profissionais da área, capacitando-os a projetar e implementar sistemas que não apenas atendam às demandas de performance e disponibilidade, mas que também sejam seguros e adaptáveis ao dinamismo do ambiente digital.

REFERÊNCIAS:

PEREIRA, Caique Silva. Sistemas distribuídos. Londrina: Editora e Distribuidora Educacional S.A., 2019. 184 p.

- MICROSOFT. Preços e licenciamento para o Windows Server 2019. Disponível em: <https://www.microsoft.com/pt-br/cloud-platform/windows-server-pricing>. Acesso em: 23 out. 2018.
- MICROSOFT. Visual Studio Professional 2017. Disponível em: <https://www.microsoft.com/pt-pt/p/visual-studio-professional-2017/dg7gmgf0dst5>. Acesso em: 23 out. 2018.
- NTP. Network Time Protocol project. [S.l.], 2018. Disponível em: <http://www.ntp.org/>. Acesso em: 18 out. 2018.
- ORACLE. CORBA Technology and the Java™ Platform Standard Edition. Disponível em: <https://docs.oracle.com/javase/7/docs/technotes/guides/idl/corba.html>. Acesso em: 8 jan. 2019.
- ORACLE. Getting Started Using Java™ RMI. Disponível em: <https://docs.oracle.com/javase/7/docs/technotes/guides/rmi/hello/hello-world.html>. Acesso em: 8 jan. 2019.
- ORACLE. Oracle VM VirtualBox®: User Manual. 2018. Disponível em: <https://www.virtu-albox.org/manual/UserManual.html>. Acesso em: 02 nov. 2018.
- ORACLE. The Java EE 6 Tutorial. Disponível em: <https://docs.oracle.com/javaee/6/tutorial/doc/bnayl.html>. Acesso em: 8 jan. 2019.

- PEREIRA, Caique Silva. Sistemas distribuídos / Caique Silva Pereira. – Londrina : Editora e Distribuidora Educacional S.A., 2019. 184 p. ISBN 978-85-522-1443-4.
- PETABRIDGE. Building Networked .NET Applications with Akka.Remote. Disponível em: <https://petabridge.com/training/akka-remoting/>. Acesso em: 8 jan. 2019.
- PREFEITURA DE PETRÓPOLIS. Supercomputador Santos Dumont poderá ser usado por empresas inovadoras. Petrópolis, 30 jan. 2018. Disponível em: <http://www.petropolis.rj.gov.br/pmp/index.php/imprensa/noticias/item/8423-supercomputador-santos-dumont-poder%C3%A1-ser-usado-por-empresas-inovadoras.html>. Acesso em: 24 out. 2018.