



Caracterização de Sistemas Distribuídos

Gustavo Zanzin Guerreiro Martins

Departamento Acadêmico de Computação – Universidade Tecnológica Federal do
Paraná (UTFPR) – 86812-460 – Campo Mourão – PR – Brasil

gustavozanzin@alunos.utfpr.edu.br

1. Na publicação 800-145 do NIST, a Computação em Nuvem é definida com 5 características essenciais, 3 modelos de serviços e 4 modelos de implementação. Fale sucintamente com suas palavras sobre cada um deles.

1.1 Características essenciais

Serviço sob demanda e autoatendimento. Refere-se à capacidade dos usuários de provisionar recursos de computação, de forma rápida e automática, sem a necessidade de intervenção manual por parte dos provedores de serviços em nuvem.

Ampla acessibilidade de rede. Refere-se à capacidade de acessar os recursos de computação em nuvem de forma ampla, através de diferentes dispositivos e plataformas.

Agrupamento de recursos. Refere-se à prática de agrupar recursos de computação, como armazenamento, redes e servidores, para servir a múltiplos usuários de forma eficiente, permitindo a otimização e o compartilhamento desses recursos entre os usuários.

Elasticidade rápida. Refere-se à capacidade de aumentar ou diminuir rapidamente os recursos de computação, como capacidade de processamento e armazenamento, conforme a demanda do usuário, permitindo uma resposta rápida às variações de carga.

Serviço mensurado. Refere-se à capacidade de medir e controlar de forma precisa o uso dos recursos de computação em nuvem, permitindo que os usuários paguem apenas pelo que utilizam, como uma espécie de "medida do serviço" prestado.

1.2 Modelos de serviços

Software como Serviço (SaaS). Os usuários têm acesso a aplicativos de *software* hospedados na nuvem, em vez de instalá-los localmente em seus dispositivos. Os usuários acessam o *software* pela Internet, e pagam pelo uso do serviço.

Plataforma como Serviço (PaaS). Os usuários têm acesso a uma plataforma de desenvolvimento de aplicativos na nuvem, que inclui ferramentas e serviços para desenvolver, testar, implantar e gerenciar aplicativos sem a necessidade de gerenciar a infraestrutura implícita da PaaS, ainda que pagando pelo uso do serviço.

Infraestrutura como Serviço (IaaS). Os usuários têm acesso a recursos de infraestrutura de computação, como servidores virtuais, armazenamento e redes, hospedados na nuvem. Os usuários podem provisionar e gerenciar esses recursos conforme necessário, sem a

necessidade de comprar ou manter hardware físico. Eles pagam pelo uso desses recursos, geralmente com base na capacidade e na quantidade de recursos consumidos.

1.3 Modelos de implementação

Nuvem privada. Infraestrutura de computação em nuvem dedicada exclusivamente a uma única organização. Ela pode ser gerenciada pela própria organização ou por um provedor de serviços de nuvem externo, e é utilizada para hospedar aplicativos e dados da organização, oferecendo os benefícios da computação em nuvem, como escalabilidade e flexibilidade, mas mantendo o controle e a segurança dos recursos na organização.

Nuvem comunitária. Infraestrutura de computação em nuvem compartilhada por várias organizações que têm interesses comuns, como requisitos de segurança, conformidade ou políticas de governo. Essas organizações colaboram para compartilhar os custos e os benefícios da nuvem comunitária, que pode ser gerenciada por uma única organização ou por um provedor de serviços de nuvem.

Nuvem pública. Infraestrutura de computação em nuvem compartilhada por várias organizações e acessada pela Internet. Os serviços em nuvem, como armazenamento, processamento e aplicativos, são oferecidos por provedores de serviços de nuvem e disponibilizados aos usuários mediante pagamento, com base no consumo.

Nuvem híbrida. Ambiente de computação em nuvem que combina nuvens públicas e privadas, permitindo que as organizações compartilhem dados e aplicativos entre essas nuvens. Isso oferece flexibilidade para as organizações, permitindo que elas mantenham alguns dados e aplicativos em uma nuvem privada, enquanto usam a nuvem pública para outros fins, como escalabilidade e acesso global.

2. A partir do artigo "A brief Introduction to distributed systems" detalhe os conceitos: *middleware*, dimensões de escalabilidade e tipos de SD.

2.1 Middleware

É uma camada de *software* que se estende por todas as máquinas de um sistema distribuído para fornecer a mesma interface para todas as aplicações e processos, mascarando as heterogeneidades entre sistemas de *software* e seus componentes para que esses diferentes componentes possam se comunicar de forma eficiente e transparente.

Um *middleware* pode oferecer diversos serviços, como gerenciamento de transações, segurança, distribuição de objetos e comunicação entre processos, permitindo o desenvolvimento de sistemas distribuídos mais flexíveis e interoperáveis.

2.2 Dimensões de escalabilidade

A escalabilidade de um sistema pode ser mensurada por pelo menos três dimensões, a saber, escalabilidade de tamanho, geográfica e administrativa.

2.2.1 Escalabilidade de tamanho

Se mais recursos e/ou usuários precisam ser suportados, provavelmente o gargalo ou as limitações estão na centralização dos serviços, mesmo que por diferentes motivos.

A título de exemplo, pode-se encontrar serviços implementados com um único servidor executando em uma única máquina, ou ainda, um grupo de servidores cooperando numa

mesma localização de forma a gerar um *cluster*. O problema desse modelo é o gargalo decorrente do aumento de requisições.

Para fins demonstrativos do exemplo de serviço implementado numa única máquina, considere as seguintes origens do gargalo:

- Capacidade computacional (CPUs)
- Capacidade de armazenamento (incluindo a taxa de transferência entre CPUs e discos)
- Rede entre o usuário e o serviço centralizado

Primeiramente, analisando a capacidade computacional, é notório que um único servidor possui determinado limite físico de processamento, isto é, há um número máximo de CPUs que podem executar instruções simultaneamente. Dessa forma, quando o número de requisições excede a capacidade de processamento do servidor, gera-se um gargalo, resultando em tempo de resposta mais longos ou até mesmo falhas no serviço.

Em segunda instância, a capacidade de armazenamento contribui para a adversidade. Caso a taxa de transferência entre as CPUs e os discos não for suficiente para lidar com o volume de dados gerado pelas requisições, o sistema pode sobrecarregar, causando, novamente, lentidão ou falhas.

Em última análise, é válido destacar a conexão de rede entre o usuário e o sistema. Sob essa ótica, se a largura de banda da rede não for adequada para suportar o tráfego gerado pelas requisições, o tempo de resposta do serviço será afetado negativamente.

Portanto, a centralização dos serviços em um único servidor ou em um *cluster* pode levar a problemas de escalabilidade de tamanho. Para lidar com isso, é necessário distribuir os serviços em vários servidores, aproveitando a capacidade de processamento e armazenamento de forma mais eficiente. Com isso, é possível melhorar a escalabilidade do sistema e garantir um melhor desempenho, mesmo diante do aumento de requisições e usuários.

2.2.2 Escalabilidade geográfica

É aquela na qual os usuários e os recursos do sistema estão geograficamente distantes, no entanto os atrasos na comunicação dificilmente são percebidos.

Nesse contexto, um dos principais problemas é o uso de comunicação síncrona. Em tal modelo de comunicação, um cliente que solicita um serviço bloqueia-o até receber uma resposta do servidor que implementa o serviço. Isso é contrastado com redes locais (LANs), onde a comunicação entre máquinas é rápida. No entanto, em sistemas de área ampla (WANs), a comunicação pode tornar-se muito mais lenta.

2.2.3 Escalabilidade administrativa

Outra questão complexa é como escalar sistemas distribuídos entre múltiplos domínios administrativos independentes. A principal necessidade é encontrar uma forma de resolver políticas conflituosas que dizem respeito ao uso e pagamento de recursos, gerenciamento e segurança.

Diante desse cenário, quando um sistema distribuído se expande para outro domínio, é necessário protegê-lo contra ataques maliciosos desse novo domínio, ao mesmo tempo em que se protege esse domínio contra potenciais ataques do sistema distribuído. Essa complexidade aumenta conforme mais domínios são envolvidos.

Por outro lado, sistemas distribuídos *peer-to-peer*, como redes de compartilhamento de arquivos, mostram que é possível escalar sem problemas administrativos significativos, uma vez que os usuários finais colaboram para manter o sistema, com pouca intervenção administrativa direta.

2.3 Tipos de Sistemas Distribuídos

O artigo aborda os seguintes tipos de sistemas distribuídos: sistemas computacionais distribuídos, sistemas de informação distribuídos e sistemas pervasivos.

2.3.1 Sistemas computacionais para tarefas de alto desempenho

A computação de alto desempenho surgiu com a ascensão de máquinas com vários processadores e com sistemas com vários computadores. Tal tecnologia proporcionou um modelo de memória compartilhada que, por sua vez, gerou grandes avanços na performance de programas nesse contexto. Entretanto, esses modelos não são escaláveis. Por isso, desenvolveu-se o chamado sistema DSM (*distributed shared-memory multicomputers*) que possibilita um processador realizar a locação de um endereço de memória em outro computador como se estivesse em memória local. Porém a ideia de DSM também precisou ser abandonada, pois esse desempenho nunca poderia atender às expectativas dos programadores, que preferem recorrer a métodos muito mais complexos e modelos de programação de passagem de mensagens com desempenho ainda melhor.

2.3.2 Sistemas de informação distribuídos

Organizações que lidam com diversas aplicações em rede enfrentam desafios de interoperabilidade. Como resultado, muitas soluções de *middleware* foram desenvolvidas para facilitar a integração dessas aplicações em um sistema de informações corporativo. À medida que as aplicações se tornaram mais sofisticadas e separadas em componentes independentes, como os componentes de banco de dados e de processamento, tornou-se claro que a integração também deveria acontecer por meio da comunicação direta entre as aplicações, impulsionando o desenvolvimento da indústria de Integração de Aplicações Empresariais (EAI).

2.3.3 Sistemas pervasivos

Com o surgimento de dispositivos móveis e embarcados, originou-se, em sistemas distribuídos, os sistemas pervasivos. Tais sistemas são caracterizados por integrar-se naturalmente ao ambiente, com uma maior fusão entre usuários e componentes do sistema. Eles são distribuídos, mas diferem dos sistemas tradicionais pela falta de uma interface dedicada, ao invés disso, utilizam sensores e atuadores para interagir com o ambiente e os usuários. A variedade de dispositivos em sistemas pervasivos, como *smartphones*, apresenta desafios únicos, como em relação à comunicação sem fio, exigindo soluções especiais para manter a transparência e a eficiência do sistema. Nesse sentido, os autores fazem uma distinção em três subtipos de sistemas pervasivos para analisar os aspectos de cada um: sistemas de computação onipresentes, sistemas móveis e redes de sensores.

3. Fale sobre um SD e discorra sucintamente como ele provê suporte para as características (heterogeneidade, abertura, segurança, concorrência, escalabilidade, transparência, tolerância a falhas) abordadas na aula.

Para tanto será considerado o banco digital Inter.

Heterogeneidade. Para lidar com as heterogeneidades presentes em um sistema bancário é considerável a utilização de padrões e de *middlewares* em suas entidades para permitir a interação entre tais dispositivos e serviços.

Abertura. Para solucionar os desafios de interoperabilizar componentes desenvolvidos por pessoas diferentes, ou ainda, a integração com outros sistemas externos, cabe à instituição estabelecer regras para a criação de documentação dos serviços, como APIs, e dos seus processos de desenvolvimento.

Segurança. O Banco Inter deveria implementar medidas de segurança robustas, como criptografia no tráfego de dados, monitoramento de atividades suspeitas para proteger seu sistema contra ameaças não só externas como também internas.

Concorrência. O sistema da instituição bancária pode implementar técnicas de controle de concorrência nos seus bancos de dados através de *locks* e transações isoladas para garantir que várias transações financeiras possam ocorrer simultaneamente sem conflitos, com os devidos cuidados ao usar mecanismos bloqueantes.

Escalabilidade. O banco pode projetar sua infraestrutura para escalar horizontalmente, adicionando mais servidores e recursos conforme a demanda aumenta, ou ainda, implementando algum mecanismo de elasticidade, garantindo que o sistema possa tratar um grande volume de transações e usuários sem comprometer o desempenho.

Transparência. É possível ocultar a complexidade da distribuição dos recursos e processos do usuário através da virtualização de recursos, onde os clientes interagem com uma camada de abstração que esconde a complexidade da distribuição dos recursos fazendo com que o sistema seja percebido como único.

Tolerância a falhas. O Banco Inter pode implementar mecanismos de tolerância a falhas, como replicação de recursos, detecção e correção de erros automatizados a fim de assegurar que o sistema continue funcionando mesmo em caso de falhas em alguns de seus componentes.

Referências

STEEN, Maarten van; TANENBAUM, Andrew S.. *A Brief Introduction to Distributed Systems*. Springerlink.com, 1-43, 2016.

MELL, P.; GRANCE, T. *The NIST Definition of Cloud Computing: Recommendations of the National Institute of Standards and Technology*. NIST Special Publication 800-145, 1-7, setembro de 2011.