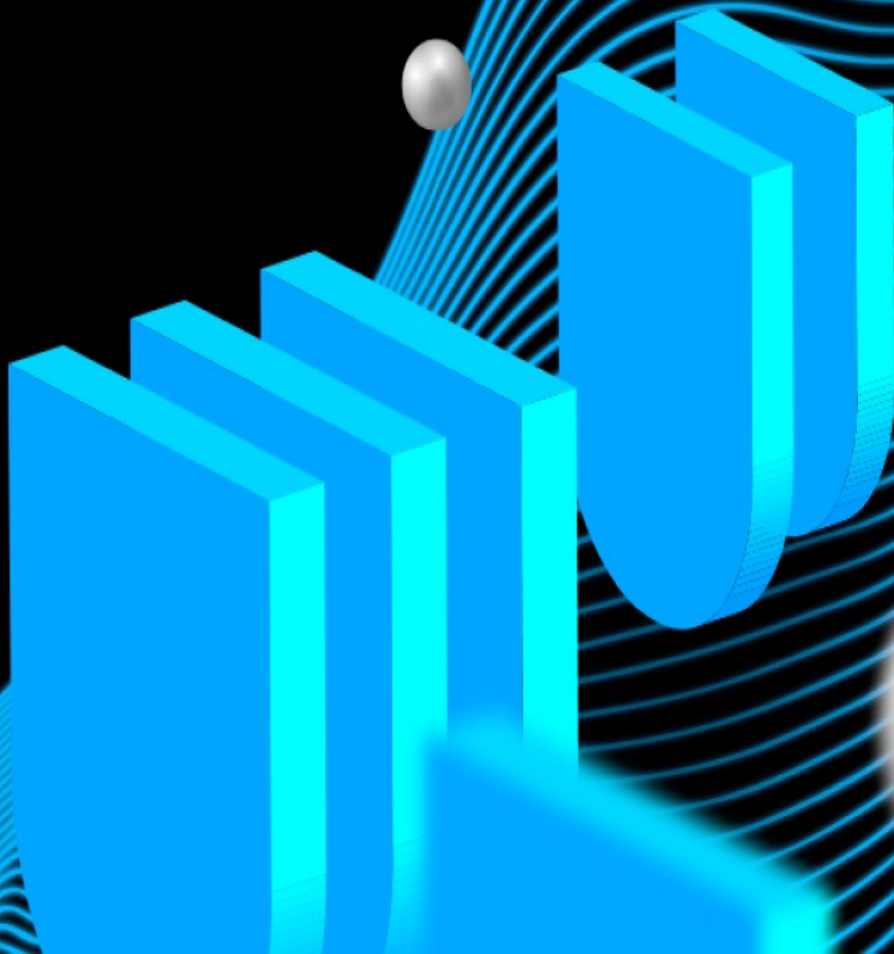




UNITAU
Universidade de Taubaté





Introdução



Introdução



As aplicações da informática tanto na área empresarial quanto na vida pessoal tem se ampliado de forma avassaladora nas últimas décadas devido ao avanços tecnológicos alcançados nesses anos.

A evolução do hardware, associado aos avanços na conectividade e a integração dos computadores a virtualmente qualquer equipamento e/ou situação do mundo moderno tem permitido o surgimento de novas aplicações e áreas de atuação da informática que seriam impossíveis até pouco tempo.

Atualmente tem grande destaque aplicações voltadas a atender um gigantesco número de usuários e, consequentemente, um gigantesco volume de operações simultâneas. A crescente importância dessas novas aplicações tem aumentado muito a importância dos sistemas distribuídos.

Introdução



Escalabilidade e tolerância a falhas passaram a ter enorme relevância para na definição de novos sistemas. Alterando a prioridade das características das tecnologias utilizadas em algumas aplicações.

Essas novas diretrizes também afetaram a área de banco de dados, tornando muito mais relevante a área de banco de dados distribuídos.

Escalabilidade



Escalabilidade é a capacidade de um sistema de hardware ou software de aumentar a carga de trabalho que o sistema consegue atender.

Para conseguir esse aumento, podem ser adotadas diferentes soluções, dependendo das características e das necessidades do sistema. Entre diversas visões e classificações possíveis para o conceito de escalabilidade, temos os conceitos de escalabilidade horizontal e escalabilidade vertical que são fatores de grande importância no projeto de um sistema.

Escalabilidade



Escalabilidade Vertical (scale out) significa adicionar mais recursos a um único nó do sistema. Para aplicações de banco de dados, significa melhorar o servidor de banco de dados, com mais memória, núcleos de processamento, disco rígido mais rápido, etc.

Escalabilidade Horizontal (scale up) significa adicionar mais nós ao sistema. Para aplicações de banco de dados, significa adicionar mais servidores a um mesmo banco de dados distribuído entre diversos servidores.

Escalabilidade



A Escalabilidade vertical tem limites determinados pelo hardware disponível. Até um certo ponto é relativamente fácil e barato fazer upgrades no hardware de um servidor. Porém, ao se aproximar dos limites da “tecnologia de ponta”, esses upgrades se tornam economicamente inviáveis. O ganho de desempenho passa a ser mínimo se comparado ao custo do upgrade.

A popularização da Internet levou a aplicações com um volume de dados e número de transações que não consegue mais ser atendida apenas com escalabilidade vertical. Essas aplicações dependem de uma utilização de clusters de servidores cada vez maiores para atender a demanda.

Escalabilidade



Os SGBDs relacionais voltados para as aplicações tradicionais tem como um dos principais enfoques a confiabilidade. A integridade das transações conforme as propriedades ACID é uma das principais características desses produtos. Porém, essa confiabilidade dificulta a escalabilidade horizontal nesses bancos.

Teorema CAP



Segundo o teorema **CAP** proposto por Eric Brewer, é impossível para um sistema de computação distribuída, atender simultaneamente a três garantias:

Consistency (consistência) - toda operação de leitura de um dado deve retornar o valor atualizado desse dados em todos os nós do sistema

Availability (disponibilidade) - os dados do sistema devem estar sempre a disposição, todas as requisições ao sistema devem receber uma resposta

Partition tolerance (tolerância a falhas de partição) - as requisições devem continuar a ser atendidas mesmo quando houver falhas na conexão entre os nós

Segundo Brewer, um sistema distribuído pode atender a duas dessas garantias ao mesmo tempo, porém não é possível atender as três garantias simultaneamente. Uma das garantias sempre deve ser sacrificada.

Teorema CAP



Sistemas que priorizam a consistência e disponibilidade (**CA) não tem grande tolerância a falhas quando a base de dados é particionada, essa é a característica dos tradicionais bancos de dados relacionais.**

Sistemas que priorizam a consistência e tolerância a falhas de particionamento (**CP) podem apresentar problemas de disponibilidade decorrentes da necessidade de garantir a consistência entre os diversos nós. Exemplos de sistemas com essas características são o BigTable do Google e o HBase.**

Teorema CAP



Sistemas que priorizam a disponibilidade e a tolerância a falhas de particionamento (**AP) deixam de lado a consistência forte do paradigma **ACID** e utilizam a consistência eventual do paradigma **BASE** (Basic Availability, Soft-state, Eventual consistency), onde após uma alteração, o sistema eventualmente atingirá um estado consistente se não houver mais alterações. A versão antiga do dado alterado continuará a existir em alguns nós até que o sistema se torne consistente. Exemplo de sistemas com essas características são os bancos Dynamo da Amazon e Cassandra.**

Introdução



As novas aplicações não substituem as aplicações tradicionais de algumas décadas atrás. São desenvolvidas para atender demandas completamente novas ou para complementar as aplicações tradicionais ampliando o alcance dessas aplicações. As novas necessidades tecnológicas não diminuem a importância das necessidades existentes anteriormente.

Por consequência, os banco de dados distribuídos podem ser voltados para atender as características já tradicionais dos bancos de dados relacionais ou podem ser voltados para atender as demandas dessas novas aplicações.

Introdução



Bancos de dados relacionais distribuídos que continuam focados nas propriedades ACID enfrentam as limitações decorrentes dessas características.

Existem bancos de dados relacionais distribuídos voltados para atender um grande volume de processamento, deixando de priorizar as transações e as propriedades ACID.

Outros modelos de bancos de dados tem surgido para atender demandas como atender a um gigantesco número de operações, priorizando a escalabilidade e tolerância a falhas.

Introdução



Iremos analisar as dificuldades existentes na implementação de bancos de dados distribuídos e estudar modelos de bancos de dados alternativos aos bancos relacionais.