UNIVERSIDADE DE BRASÍLIA
Faculdade do Gama

Sistemas de Banco de Dados 2

**Tecnologias de Banco de Dados (TI-BD)**

# Bancos de Dados Distribuídos

**Kess Jhones Gomes Tavares - 180124498**

Brasília, DF
2023

## Banco de dados distribuído

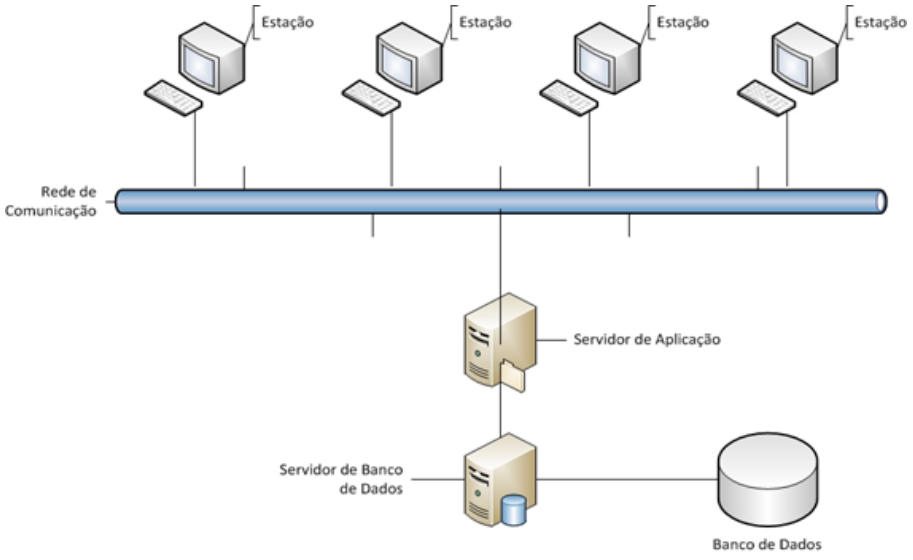
De acordo com Elmasri e Navathe (2011), um banco de dados distribuído (BDD) é composto por múltiplos bancos de dados inter-relacionados, distribuídos em uma rede de computadores, e seu sistema de gerenciamento (SGBDD) é um software que administra o banco de dados distribuído, tornando sua distribuição transparente ao usuário.

O objetivo principal de um BDD é melhorar o desempenho, a escalabilidade e a confiabilidade do armazenamento e gerenciamento de dados em ambientes de negócios complexos e em constante mudança. Neste relatório, exploraremos os principais aspectos de um BDD, incluindo sua arquitetura, principais características, vantagens e desvantagens, bem como suas aplicações.

## Arquitetura

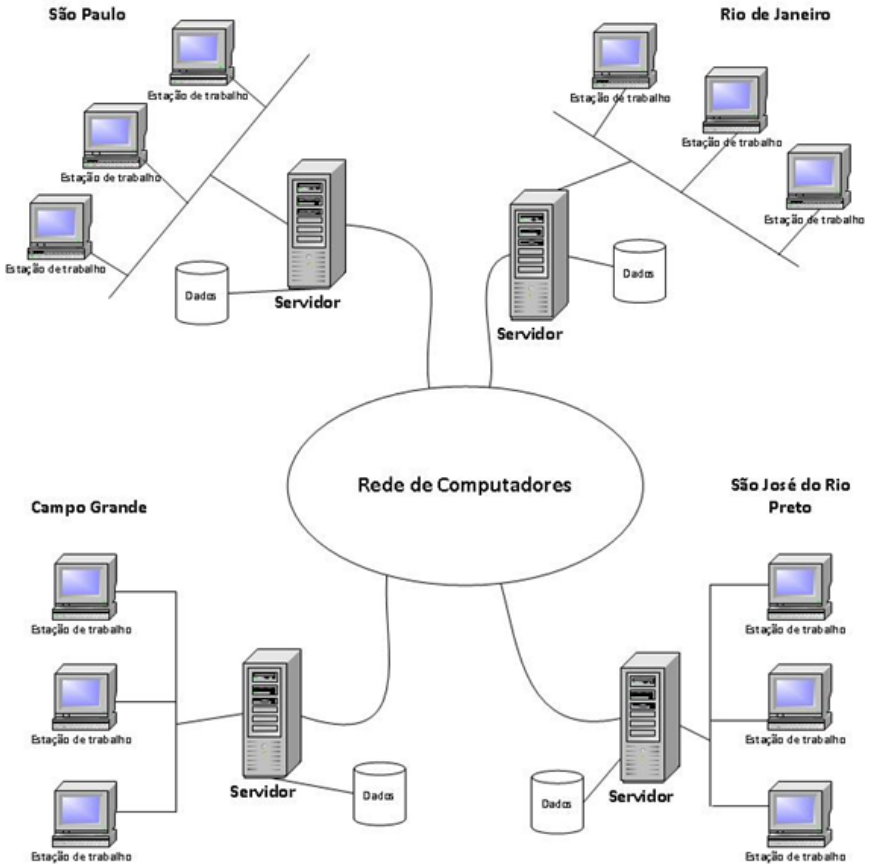
A arquitetura de um BDD é composta por uma rede de computadores interconectados, que inclui pelo menos dois nós, podendo ser local ou através da internet, cada um dos quais contém um SGBD. Ao contrário do modelo convencional de banco de dados, também chamado de centralizado, onde todas as informações são armazenadas em um único local.

**Figura 1:** Arquitetura de Banco de Dados Centralizado




**Fonte:** Publicação sobre BDD no site imaster

**Figura 2:** Arquitetura de Banco de Dados Distribuídos




**Fonte:** Publicação sobre BDD no site imaster

Existem várias formas de implementação de um BDD, incluindo a replicação de dados e fragmentação de dados. A replicação de dados envolve a cópia dos dados em vários nós para aumentar a disponibilidade dos dados e melhorar a capacidade de recuperação de falhas. Porém em casos onde a replicação é total, onde se tem a cópia do banco de dados inteiro em cada nó, a operação de atualização pode ser bastante atrasada. (ELMASRI; NAVATHE, 2011).

A fragmentação de dados envolve a divisão dos dados em fragmentos menores, que são distribuídos entre os nós para melhorar o desempenho e a escalabilidade. Existem várias técnicas de particionamento de dados, incluindo particionamento horizontal, particionamento vertical e particionamento híbrido.

A fragmentação horizontal envolve a divisão dos dados em linhas, onde cada nó é responsável por um subconjunto de tuplas. Como na tabela contida

na figura 3 onde podemos determinar que teremos um nó para cada estado brasileiro e assim somos capazes de particionar a relação com base no atributo estado, um exemplo de como ficaria o nó SP está presente na figura 4.

**Figura 3:** Tabela ENDERECO

| idEndereco | cep | estado | cidade | bairro | numero | complemento | latitude | longitude |
|---|---|---|---|---|---|---|---|---|
| 1 | 68700216 | PA | Capanema | Igrejinha | 200 | Terreo 10 | -15.875934 | -48.090278 |
| 2 | 60170001 | CE | Fortaleza | Aldeota | 205 | Anexo 8 | -12.124375 | -43.893500 |
| 3 | 4302020 | SP | São Paulo | Parque Imperial | 5 | Subsolo 7 | -13.146581 | -51.766289 |
| 4 | 64000290 | PI | Teresina | Centro | 98 | Sala 1 | -12.137577 | -54.872674 |
| 5 | 79002290 | MS | Campo Grande | Monte Castelo | 8 | Apartamento 4 | -14.484110 | -55.240716 |
| 6 | 75802095 | GO | Jataí | Vila Jardim Rio Claro | 24 | Galpão 8 | -9.862212 | -53.389519 |
| 7 | 29946490 | ES | São Mateus | Guriri Norte | 12 | Sobreloja 3 | -11.320057 | -59.289178 |
| 8 | 28035042 | RJ | Campos dos Goytacazes | Centro | 6 | Anexo 4 | -7.557072 | -56.751336 |
| 9 | 30130005 | MG | Belo Horizonte | Boa Viagem | 35 | Anexo 10 | -6.557072 | -66.751336 |
| 10 | 96204040 | RS | Rio Grande | Zona Portuária | 31 | Bloco 7 | -8.125521 | -59.628855 |
| 11 | 4545005 | SP | São Paulo | Vila Olímpia | 5891 | Lote 10 | -0.660205 | -54.728953 |
| 12 | 64000291 | PI | Teresina | Centro | 8891 | Sobreloja 10 | -10.380788 | -42.687936 |
| 13 | 20040002 | RJ | Rio de Janeiro | proximo ao estadio | 8732 | Terreo 4 | -24.179821 | -53.312843 |
| 14 | 13216000 | SP | Jundiaí | Vila Joana | 2530 | Terreo 3 | -24.280006 | -49.896095 |
| 15 | 88804115 | PA | Belém | Nazaré | 3544 | Conjunto 7 | -28.839600 | -51.862677 |

**Fonte:** produção do autor

**Figura 4:** Fragmentação horizontal

| idEndereco | cep | estado | cidade | bairro | numero | complemento | latitude | longitude |
|---|---|---|---|---|---|---|---|---|
| 3 | 4302020 | SP | São Paulo | Parque Imperial | 5 | Subsolo 7 | -13.146581 | -51.766289 |
| 11 | 4545005 | SP | São Paulo | Vila Olímpia | 5891 | Lote 10 | -0.660205 | -54.728953 |
| 14 | 13216000 | SP | Jundiaí | Vila Joana | 2530 | Terreo 3 | -24.280006 | -49.896095 |
| NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL | NULL |

**Fonte:** produção do autor

A fragmentação vertical envolve a divisão dos dados em colunas, onde cada nó é responsável por um subconjunto de colunas. Novamente utilizando a figura 3, podemos particionar verticalmente a tabela, se por exemplo definirmos que apenas os atributos cep, estado e cidades são necessários em um determinado nó, resultado na figura 5. O particionamento híbrido envolve uma combinação dos dois métodos anteriores.

**Figura 4:** Fragmentação vertical

| idEndereco | cep | estado | cidade |
|---|---|---|---|
| 1 | 68700216 | PA | Capanema |
| 2 | 60170001 | CE | Fortaleza |
| 3 | 4302020 | SP | São Paulo |
| 4 | 64000290 | PI | Teresina |
| 5 | 79002290 | MS | Campo Grande |
| 6 | 75802095 | GO | Jataí |
| 7 | 29946490 | ES | São Mateus |
| 8 | 28035042 | RJ | Campos dos Goytacazes |
| 9 | 30130005 | MG | Belo Horizonte |
| 10 | 96204040 | RS | Rio Grande |
| 11 | 4545005 | SP | São Paulo |
| 12 | 64000291 | PI | Teresina |
| 13 | 20040002 | RJ | Rio de Janeiro |
| 14 | 13216000 | SP | Jundiaí |
| 15 | 88804115 | PA | Belém |

**Fonte:** produção do autor

Independente de qual forma for utilizada para a implementação, em um banco de dados distribuído, a arquitetura é projetada de forma a ser transparente para os usuários que utilizam as aplicações, ou seja, para o usuário final, os dados parecem estar centralizados em um único servidor, embora, na verdade, eles possam estar distribuídos em vários locais fisicamente separados (IMASTERS, 2013).

## Desvantagens

Os bancos de dados distribuídos apresentam algumas desvantagens em relação aos bancos de dados centralizados ou localizados em um único local. Algumas dessas desvantagens incluem:

- **Complexidade de gerenciamento:** a administração de um banco de dados distribuído pode ser mais complexa do que a de um banco de dados centralizado. Isso ocorre porque os dados são armazenados em diferentes locais e podem ser acessados por vários usuários simultaneamente, exigindo uma coordenação e um controle de acesso

mais rigorosos (IMASTERS, 2013).

- **Dependência da rede:** os bancos de dados distribuídos dependem da rede para trocar dados e sincronizar transações entre os diferentes locais. Isso significa que qualquer problema na rede, como interrupções ou latência, pode afetar o desempenho e a disponibilidade do banco de dados.

- **Diferenças nos modelos de dados:** Em um BDD, os bancos de dados podem ser provenientes de diversos modelos de dados, incluindo modelos legados, modelo relacional, não relacionais e etc. No entanto, a capacidade de modelagem pode variar e tornar desafiador o processamento dos modelos de forma uniforme por meio de um único esquema global ou em uma única linguagem. Mesmo que os bancos de dados sejam do mesmo ambiente de SGBDR, a mesma informação pode ser representada de forma diferente, exigindo um mecanismo inteligente de processamento de consulta que possa relacionar as informações com base nos metadados (ELMASRI; NAVATHE, 2011).

- **Segurança:** "As transações distribuídas precisam ser executadas com o gerenciamento apropriado da segurança dos dados e dos privilégios de autorização/acesso dos usuários." (ELMASRI; NAVATHE, 2011, p. 593). Assim a complexidade de segurança aumenta de acordo com o número de nós, pois tem que se garantir os aspectos de segurança em todos os nós.

É importante levar em consideração essas desvantagens ao decidir se um banco de dados distribuído é adequado para um determinado cenário. É necessário avaliar cuidadosamente os custos e benefícios, bem como as necessidades específicas de armazenamento e acesso aos dados antes de optar por um banco de dados distribuído.

## Vantagens

Os bancos de dados distribuídos apresentam diversas vantagens em relação aos bancos de dados centralizados ou localizados em um único local. Algumas das principais vantagens incluem:

- **Desempenho:** os bancos de dados distribuídos podem apresentar um melhor desempenho do que os bancos de dados centralizados, especialmente em ambientes onde os dados são acessados por um grande número de usuários ou em cenários com grande volume de dados. Isso ocorre porque os dados podem ser armazenados em vários servidores, permitindo que as consultas sejam distribuídas e executadas em paralelo.

- **Disponibilidade:** os bancos de dados distribuídos oferecem maior disponibilidade em comparação com os bancos de dados centralizados, pois os dados podem ser replicados em vários servidores, garantindo que, mesmo que ocorra uma falha em um dos servidores, os dados ainda estarão disponíveis.

- **Escalabilidade:** os bancos de dados distribuídos são mais escaláveis do que os bancos de dados centralizados, pois permitem que os dados sejam distribuídos em vários servidores. Isso significa que, à medida que a quantidade de dados aumenta, é possível adicionar mais servidores para lidar com o aumento do volume de dados.

- **Tolerância a falhas:** os bancos de dados distribuídos são mais tolerantes a falhas do que os bancos de dados centralizados, pois se for implementado replicação de dados, permite que os dados sejam replicados em vários servidores. Isso significa que, mesmo que ocorra uma falha em um dos servidores, os dados ainda estarão disponíveis em outros servidores.

Em geral, os bancos de dados distribuídos podem oferecer várias vantagens em relação aos bancos de dados centralizados, especialmente em ambientes onde os dados são acessados por um grande número de usuários ou em cenários com grande volume de dados. No entanto, é importante levar em consideração as desvantagens e avaliar cuidadosamente os custos e benefícios antes de optar por um banco de dados distribuído.

**Apache Cassandra**

O Apache Cassandra é um sistema de gerenciamento de bancos de dados distribuído e open-source, que foi desenvolvido inicialmente pelo Facebook para lidar com grandes volumes de dados em ambientes distribuídos. Em 2008, o Facebook liberou o código-fonte do Cassandra sob a licença Apache 2.0, tornando-o um projeto de código aberto. Desde então, o projeto tem sido mantido e desenvolvido pela Apache Software Foundation.

O Cassandra foi projetado para ser altamente escalável e tolerante a falhas, permitindo que grandes quantidades de dados sejam armazenadas e consultadas de maneira eficiente e confiável, mesmo em ambientes distribuídos. Ele é utilizado em diversas aplicações e serviços, como o Twitter, o Reddit e o Spotify, entre outros.

O Instagram também utiliza o Cassandra e em 2018 eles trabalhavam com uma taxa de falha da solicitação inferior a 0,001% e um tempo médio de latência de 20 ms.

**HBase**

Em resumo, o HBase é um banco de dados distribuído open-source orientado à coluna, que foi modelado a partir do Google BigTable e implementado em Java. Uma das principais vantagens do HBase é a sua fácil integração com o Hadoop, permitindo que ele utilize o MapReduce para distribuir o processamento dos dados, possibilitando assim o processamento de grandes quantidades de dados, chegando a casa dos terabytes, de forma eficiente

## Bibliografia

MESQUITA, Eduardo José Soler; FINGER, Marcelo. Projeto de Dados em Bancos de Dados Distribuídos. 1998. 206 f. Tese (Doutorado em Ciência da Computação e Matemática Computacional) - Instituto de Matemática e Estatística, Universidade de São Paulo, São Paulo, 1998.

ELMASRI, Ramez; NAVATHE, Shamkant B. Sistemas de Banco de Dados. 6. ed. Tradução de Daniel Vieira. São Paulo: Pearson Addison Wesley, 2011.

IMASTERS. O que é banco de dados distribuído?(2013). Disponível em:https://imasters.com.br/banco-de-dados/o-que-e-banco-de-dados-distribuido/?trace=1519021197&source=single%20Acesso. Acesso em: 14 de abril de 2023.

Apache Software Foundation. Apache Cassandra case studies. Disponível em: https://cassandra.apache.org/_/case-studies.html. Acesso em: 17 de abril de 2023.

Instagram Engineering. Open Sourcing a 10x Reduction in Apache Cassandra Tail Latency. Disponível em: https://instagram-engineering.com/open-sourcing-a-10x-reduction-in-apache-cassandra-tail-latency-d64f86b43589. Acesso em: 17 de abril de 2023.

The Apache HBase website. Disponível em: https://hbase.apache.org/. Acesso em: 17 de abril de 2023.