

27/02/2025

04



Professor: Moisés Silva de Sousa

O QUE VEREMOS?



Introdução

O que vimos até aqui?



Linguagens e Protocolos

Os bancos de dados seguem regra pré-estabelecidas



Extras

Big Data e Cloud



Modelos de BD

Podem ser relacionais ou não relacionais



Tecnologias para Alta Performance

Como lidar com uma carga de dados maior?



MODELOS DE BANCO DE DADOS

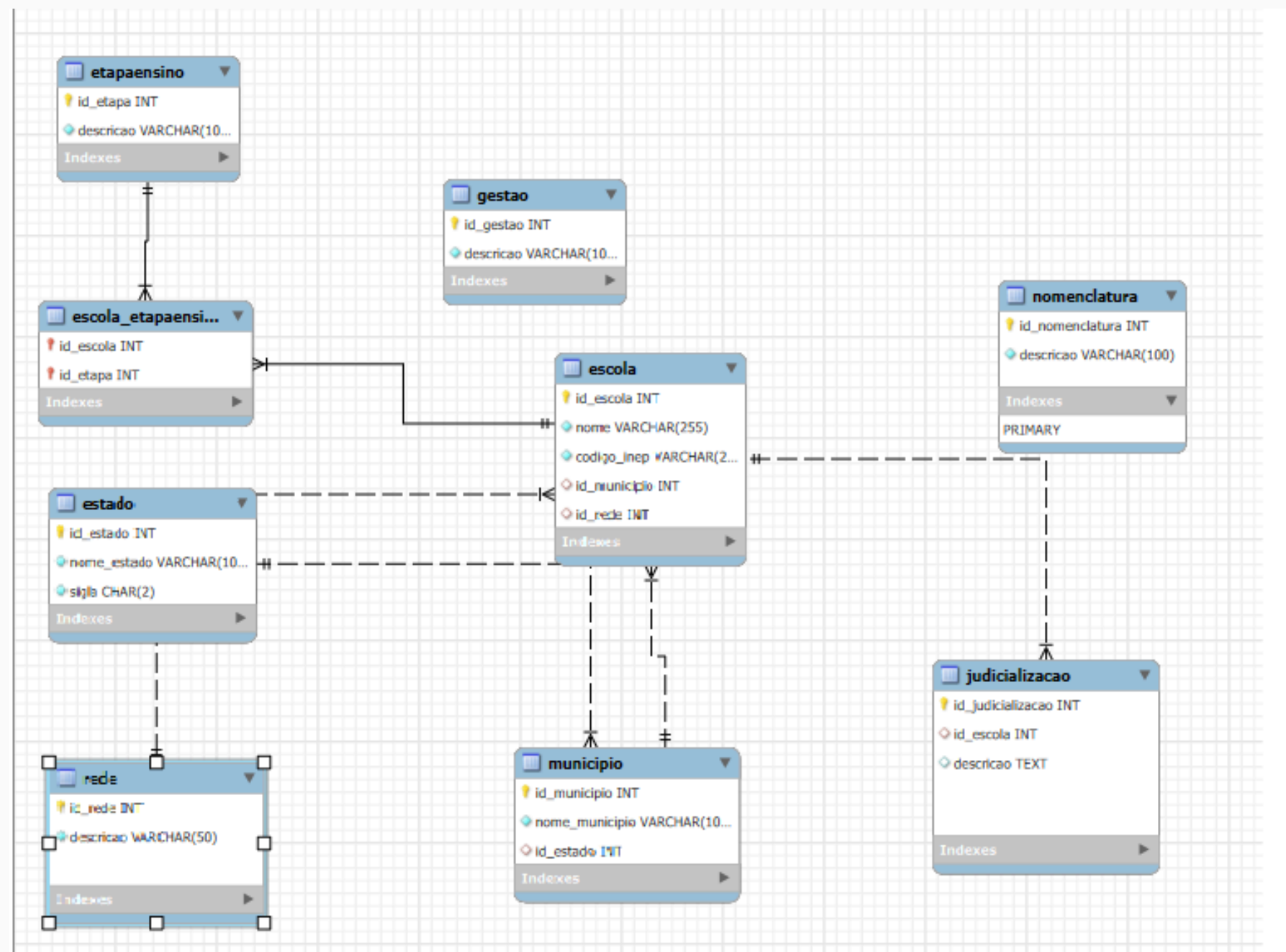
Relacional (SQL): Organiza os dados em tabelas interligadas por chaves primárias e estrangeiras. Exemplos: MySQL, PostgreSQL, SQL Server, Oracle.

NoSQL: Projetado para trabalhar com grandes volumes de dados não estruturados ou semiestruturados. Tipos:

- Documentos: MongoDB, CouchDB.
- Chave-Valor: Redis, DynamoDB.
- Colunar: Apache Cassandra, HBase.
- Grafos: Neo4j, ArangoDB.

LINGUAGENS E PROTOCOLOS

SQL (Structured Query Language): Linguagem padrão para bancos relacionais.



LINGUAGENS E PROTOCOLOS

JSON: Utilizados em bancos de documentos como MongoDB.

```
[{
  "time": "Corinthians",
  "estadio": {
    "nome": "Neo Química Arena",
    "publico_recorde": 63267,
    "capacidade": null
  },
  "jogadores": [{
    "nome": "Cássio",
    "idade": 33,
    "titular": true,
    "times": ["Grêmio", "PSV", "Corinthians"]
  },
  {
    "nome": "Luan",
    "idade": 27,
    "titular": false,
    "times": ["Grêmio", "Corinthians"]
  }
}]
```

LINGUAGENS E PROTOCOLOS

Modos de Conexão

- TCP/IP → Usado para conexões remotas (porta padrão **3306**).
- Unix Socket → Usado em sistemas Unix/Linux para comunicação local mais rápida (exemplo: `/var/run/mysqld/mysqld.sock`).

LINGUAGENS E PROTOCOLOS

- SSL/TLS (Secure Sockets Layer / Transport Layer Security) – Criptografa a comunicação entre cliente e servidor para evitar ataques de interceptação.
- LDAP (Lightweight Directory Access Protocol) – Permite autenticação baseada em diretórios centralizados como Active Directory.
- OAuth – Utilizado para autenticação segura via tokens, especialmente em bancos de dados na nuvem.

LINGUAGENS E PROTOCOLOS

- MySQL Replication Protocol (Binlog Replication) – Replica alterações do servidor mestre para os servidores secundários via log binário.

TECNOLOGIAS PARA ALTA PERFORMANCE

Sharding (Particionamento Horizontal)

Técnica para dividir um banco de dados em várias partes (shards) e distribuí-las entre diferentes servidores.

Evita sobrecarga em um único servidor e melhora a escalabilidade horizontal.

Exemplos de bancos que suportam sharding:

- MySQL (MySQL Cluster, Vitess)
- MongoDB (Sharded Cluster)
- PostgreSQL (Citus)
- Cassandra (Data Partitioning).

TECNOLOGIAS PARA ALTA PERFORMANCE

Exemplo de Sharding

Um e-commerce com milhões de usuários pode dividir sua base de dados assim:

- **Usuários com ID de 1 a 1M → Servidor A**
- **Usuários com ID de 1M a 2M → Servidor B**
- **Usuários com ID de 2M a 3M → Servidor C**

Isso reduz a carga em cada servidor e melhora o desempenho.

TECNOLOGIAS PARA ALTA PERFORMANCE

Replication (Replicação de Dados)

- Mantém cópias de um banco de dados em múltiplos servidores para melhorar disponibilidade, redundância e escalabilidade.
- Tipos de replicação:
 - a. Master-Slave: Um servidor principal (master) replica as alterações para um ou mais servidores secundários (slaves).
 - b. Master-Master: Vários servidores podem aceitar gravações, permitindo balanceamento de carga.
 - c. Multimaster: Usado em bancos NoSQL e distribuídos para alta escalabilidade.
 - d. Logical Replication: Apenas determinadas tabelas ou registros são replicados (PostgreSQL).
 - e. Binlog Replication: No MySQL, usa logs binários para replicação assíncrona.

TECNOLOGIAS PARA ALTA PERFORMANCE

Caching (Armazenamento em Memória)

O cache melhora a velocidade das consultas armazenando os dados mais acessados na memória.

1. Query Cache (MySQL)

- Armazena resultados de consultas SQL repetitivas para acelerar respostas.
- Deprecado no MySQL 8.0 devido a limitações de escalabilidade.

2. CDN + Database Caching

- Utiliza Cloudflare, AWS CloudFront, Varnish para armazenar respostas de banco de dados e reduzir chamadas diretas ao SGBD.