

História e evolução dos Bancos de Dados

Prof. Esp. Claudio Lopes Ferrini Garcia

Banco de Dados
Análise e Desenvolvimento de Sistemas
Fatec Taubaté

Objetivos da Apresentação

- Compreender a evolução histórica dos bancos de dados
- Identificar as principais gerações e modelos de bancos de dados
- Conhecer os modelos conceituais de informações
- Relacionar evolução tecnológica e organização dos dados

História e evolução dos Bancos de Dados

- Antes dos computadores: registros em papel (arquivos manuais)
- Primeiras aplicações na década de 1960
- Uso de sistemas de arquivos para armazenar dados digitalmente
- Limitações: redundância, dificuldade de atualização, falta de integridade

Linha do Tempo resumida

- Anos 1960 – Sistemas hierárquicos e de rede (ex.: IMS da IBM, IDMS)
- Anos 1970 – Surgimento do modelo relacional (proposto por Edgar F. Codd)
- Anos 1980 – Expansão dos SGBDs relacionais comerciais (Oracle, DB2, SQL Server)
- Anos 1990 – Data Warehousing e OLAP
- Anos 2000 – Bancos de dados orientados a objetos e XML
- Anos 2010+ – Big Data, NoSQL, bancos em nuvem
- Atualidade – Bancos híbridos (NewSQL, multimodelo), IA e automação de consultas

Banco de Dados Relacional

- tipo de banco de dados que organiza os dados em tabelas com linhas e colunas,
- cada linha representa um registro e cada coluna representa um atributo dos dados

Banco de Dados Relacional

Estrutura tabular:

- Dados são armazenados em tabelas, com linhas (registros) e colunas (atributos).

Relacionamentos:

- Tabelas podem ser relacionadas através de chaves primárias e estrangeiras, permitindo a conexão entre diferentes conjuntos de dados.

Linguagem SQL:

- Bancos de dados relacionais utilizam a linguagem SQL para consulta e manipulação dos dados.

Banco de Dados Relacional

Integridade referencial:

- O modelo garante a consistência e a precisão dos dados, evitando inconsistências e erros.

Transações ACID:

- Suporte a transações com propriedades de Atomicidade, Consistência, Isolamento e Durabilidade (ACID) para garantir a integridade das operações.

Banco de Dados Relacional - Vantagens

- Organização e flexibilidade
- Integridade dos Dados
- Escalabilidade Vertical (capacidade de armazenamento)
- Facilidade de uso

Banco de Dados Relacional - Desvantagens

- Escalabilidade horizontal (distribuição dos dados em vários servidores)
- Rigidez do esquema de dados
- Complexidade
- Custos
- Dificuldade de adaptação a sistemas de dados não estruturados

Relação entre Data Warehousing e OLAP:

Complementaridade:

O DW fornece o repositório de dados estruturado e otimizado para análise, enquanto o OLAP oferece as ferramentas para analisar esses dados de forma interativa e multidimensional.

Integração:

Um data warehouse bem projetado deve levar em consideração as necessidades e requisitos da análise OLAP.

Relação entre Data Warehousing e OLAP:

Eficiência:

A combinação de DW e OLAP permite análises rápidas e eficientes de grandes volumes de dados, auxiliando na tomada de decisões estratégicas.

Em resumo: Data warehousing e OLAP são tecnologias essenciais para análise de dados em ambientes corporativos. O DW armazena e organiza os dados, enquanto o OLAP oferece ferramentas para explorá-los de forma interativa e multidimensional.

Banco de Dados Orientado à Objetos

Vantagens:

- Permitem representar dados complexos de forma natural, refletindo a estrutura do mundo real.
- Facilitam a manipulação de dados através de operações orientadas a objetos.
- Podem ser mais eficientes para certos tipos de aplicações que lidam com estruturas de dados complexas.

Banco de Dados Orientado à Objetos

Desvantagens:

- Complexidade (implementação e gerenciamento)
- Falta de Padrozinção e Maturidade
- Baixo desempenho (consultas complexas)
- Custos
- Concorrências e opções no mercado
- Segurança (poucos mecanismos e protocolos desenvolvidos)

NoSQL

bancos de dados que não utilizam o modelo relacional tradicional para armazenar informações, mas sim outros formatos de dados, como chave-valor, documentos, colunas ou grafos

NoSQL - Características

Flexibilidade de esquema:

Diferente dos bancos de dados relacionais, muitos bancos NoSQL não exigem um esquema rígido, permitindo a inclusão de dados com diferentes estruturas no mesmo conjunto.

Escalabilidade horizontal:

NoSQL bancos de dados são projetados para escalar horizontalmente, adicionando mais máquinas para lidar com o aumento de dados e tráfego, em vez de depender apenas da capacidade de uma única máquina.

NoSQL - Características

Desempenho otimizado:

Bancos de dados NoSQL são otimizados para diferentes modelos de dados e padrões de carga de trabalho, como chave-valor, colunas largas ou armazenamento em memória.

Consistência flexível:

Enquanto os bancos de dados relacionais geralmente buscam consistência forte (todos os dados devem ser consistentes em todas as réplicas), muitos bancos NoSQL oferecem opções de consistência eventual, permitindo respostas mais rápidas com possíveis dados ligeiramente desatualizados.

NoSQL - Vantagens

Flexibilidade e escalabilidade:

- Ideais para aplicações que lidam com grandes volumes de dados e precisam de flexibilidade e escalabilidade.

Desempenho em grandes volumes:

- Otimizados para lidar com grandes volumes de dados e consultas complexas.

Adaptabilidade:

- Bancos NoSQL podem ser adaptados para diferentes necessidades, oferecendo opções de consistência, modelo de dados e escalabilidade.

NoSQL - Desvantagens

Consistência:

- Alguns bancos NoSQL podem sacrificar a consistência forte para obter maior desempenho e escalabilidade.

Complexidade:

- A escolha e configuração de um banco NoSQL pode ser mais complexa do que um banco de dados relacional tradicional.

Suporte a transações:

- Alguns bancos NoSQL podem não oferecer suporte a transações complexas ou suporte limitado a transações entre diferentes partes do banco

Big Data

- conjuntos de dados extremamente grandes e complexos que são difíceis de processar usando ferramentas tradicionais de gerenciamento de dados

Big Data

Volume:

- Refere-se à quantidade massiva de dados gerados e armazenados, muitas vezes medidos em terabytes, petabytes ou exabytes.

Variedade:

- Representa a diversidade de tipos de dados, incluindo dados estruturados (como bancos de dados relacionais), não estruturados (como texto, imagens, vídeos) e semiestruturados (como arquivos JSON).

Velocidade:

- Indica a rapidez com que os dados são gerados e precisam ser processados para que as informações relevantes sejam identificadas a tempo.

Big Data - Importância

- Tomada de Decisão Aprimorada
- Melhoria da experiência do cliente
- Inovação e desenvolvimento de novos produtos
- Otimização de processos
- Prevenção de fraudes e riscos

NewSQL

NewSQL é uma classe de sistemas de gerenciamento de banco de dados que combina a escalabilidade dos sistemas NoSQL com as garantias ACID dos sistemas relacionais tradicionais

NewSQL - Vantagens

Escalabilidade:

Assim como o NoSQL, o NewSQL permite a escalabilidade horizontal, ou seja, a capacidade de distribuir os dados e o processamento em múltiplos servidores ou nós, o que o torna adequado para lidar com grandes volumes de dados e alto tráfego.

NewSQL - Vantagens

Garantias ACID:

Ao contrário do NoSQL, que pode sacrificar a consistência para alcançar alta escalabilidade, o NewSQL mantém as garantias ACID (Atomicidade, Consistência, Isolamento e Durabilidade), assegurando a integridade e confiabilidade das transações.

SQL como linguagem:

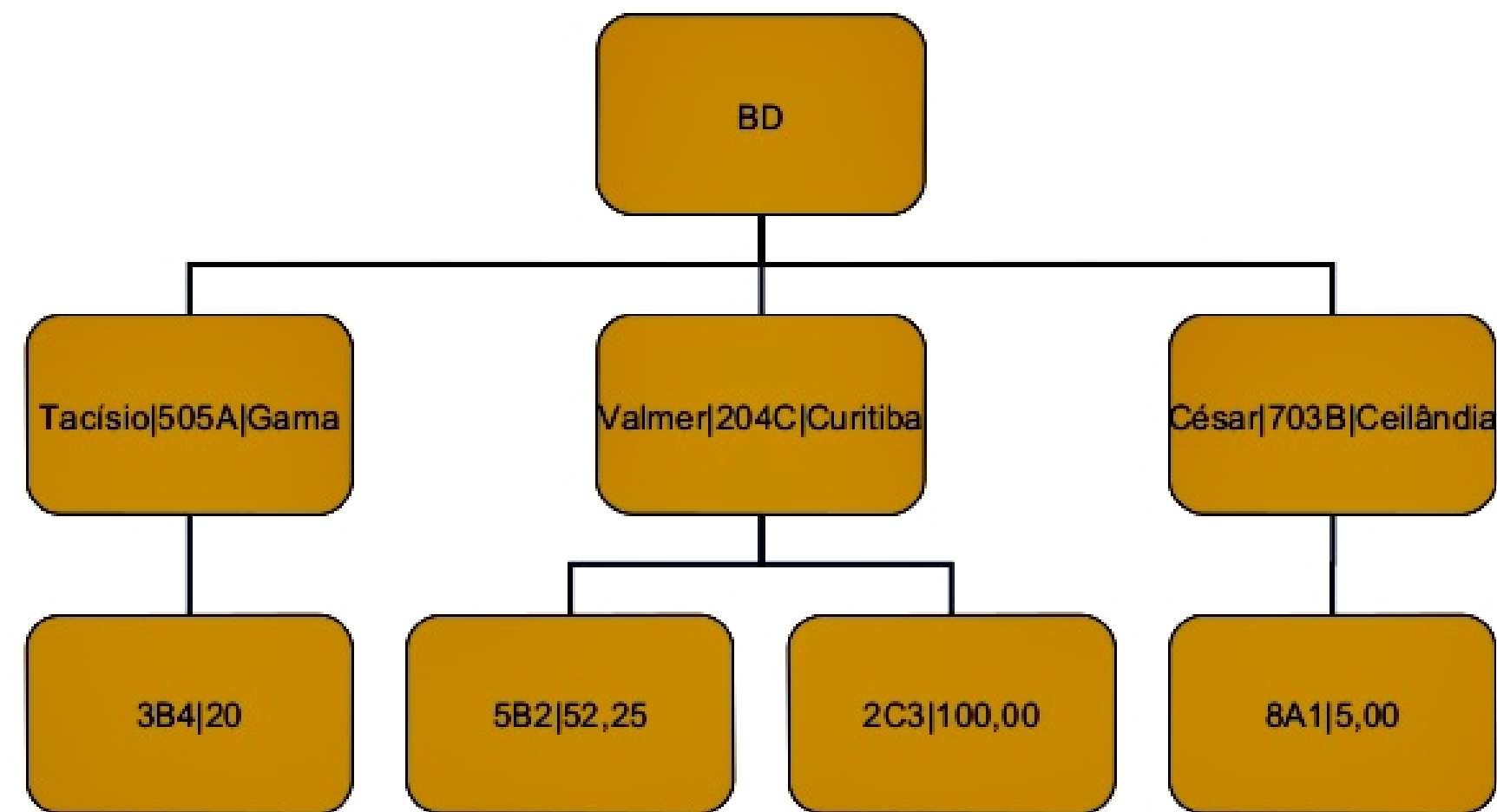
O NewSQL utiliza a linguagem SQL para consultas e manipulação de dados, tornando a transição de sistemas tradicionais mais fácil e familiar para os desenvolvedores.

Ex. Google Cloud Spanner

Evolução dos Modelos

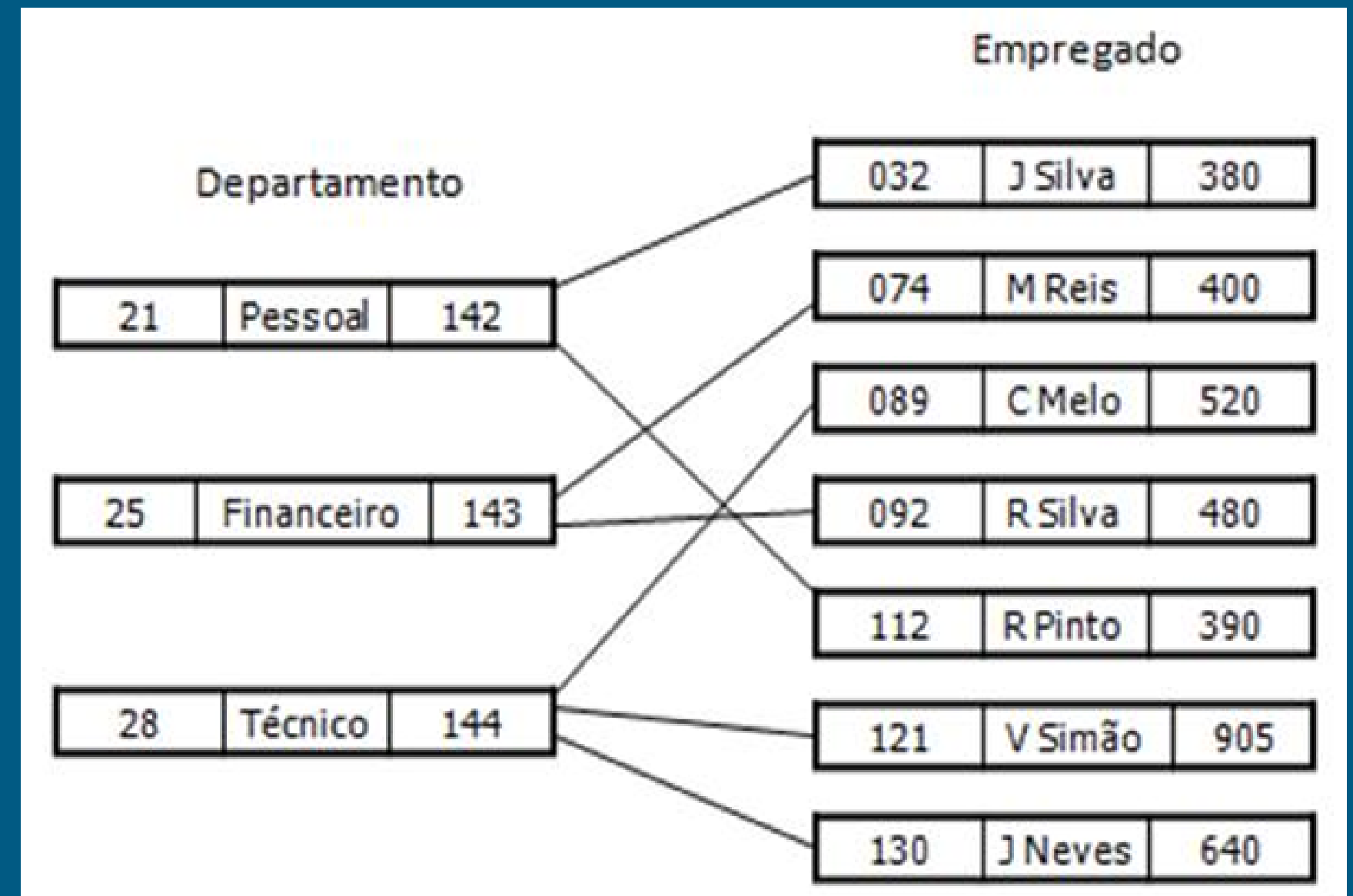
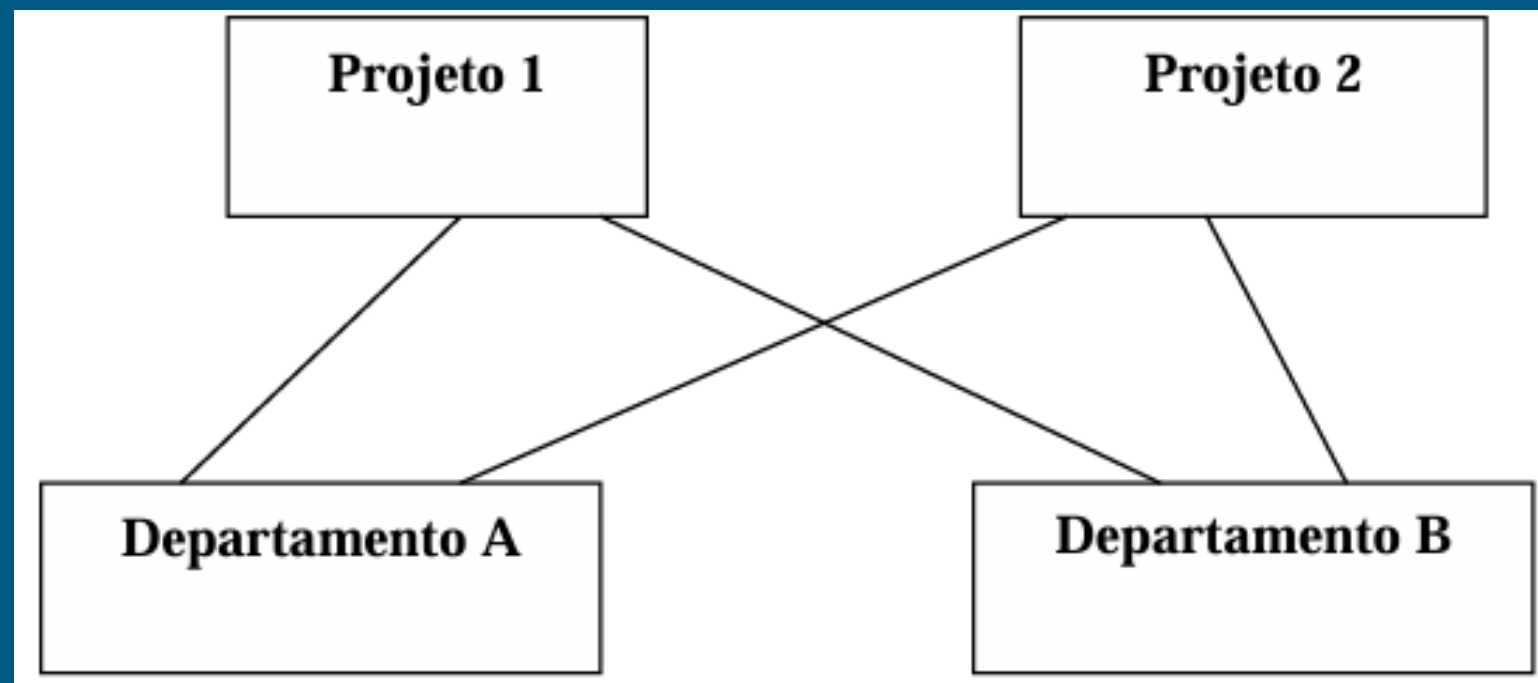
- Hierárquico – Estrutura em árvore (pai/filho)

Modelo de Dados Hierárquicos



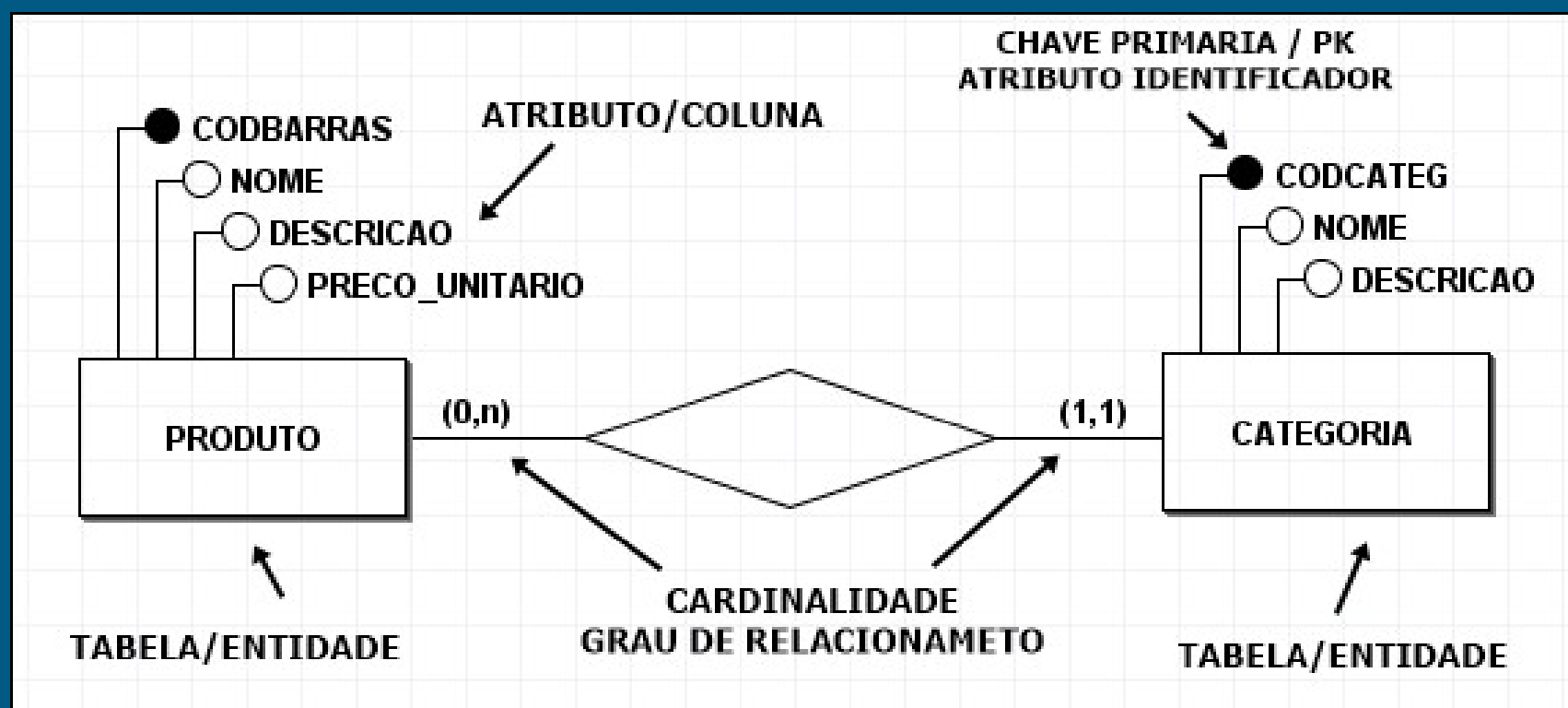
Evolução dos Modelos

Rede – Estrutura mais flexível com múltiplos relacionamentos



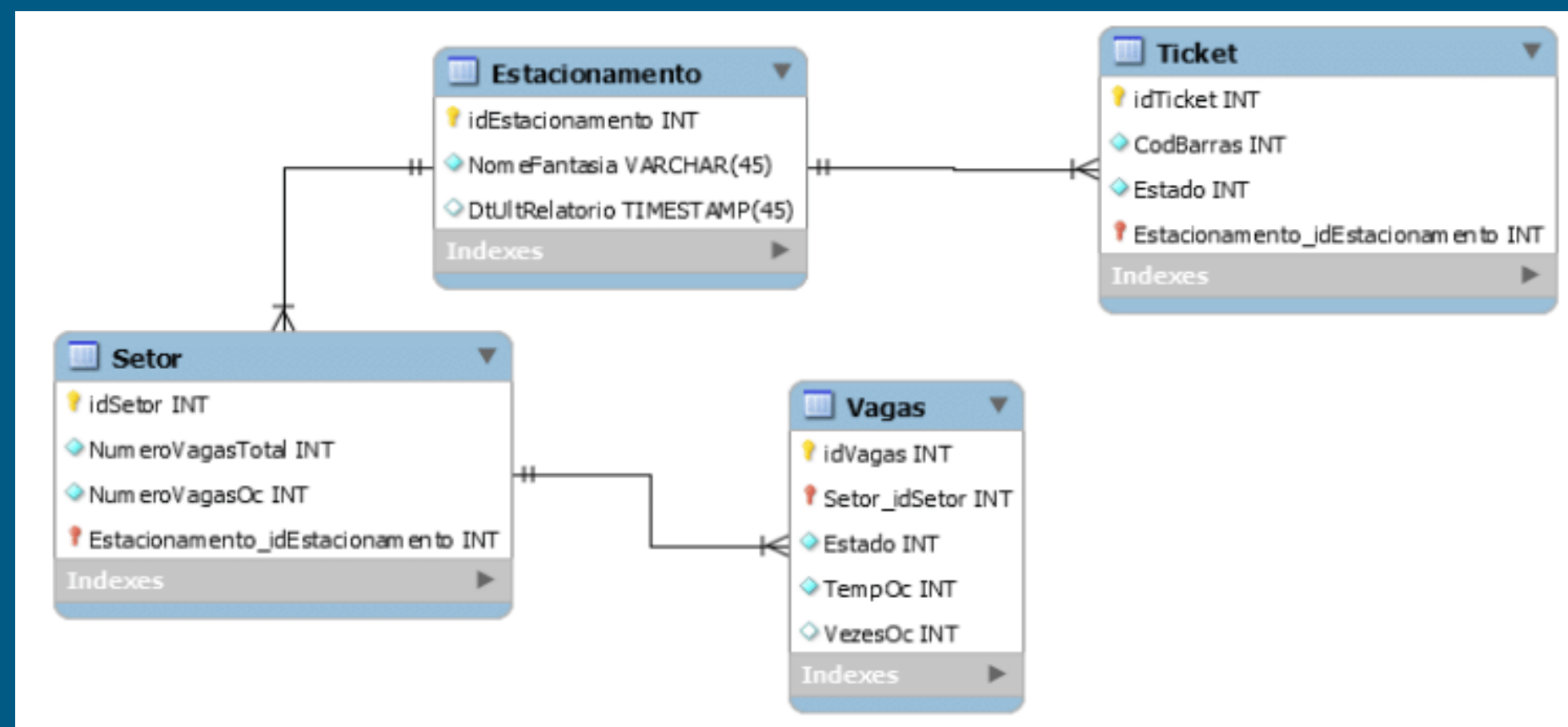
Evolução dos Modelos

Relacional – Tabelas e uso de SQL



fonte:

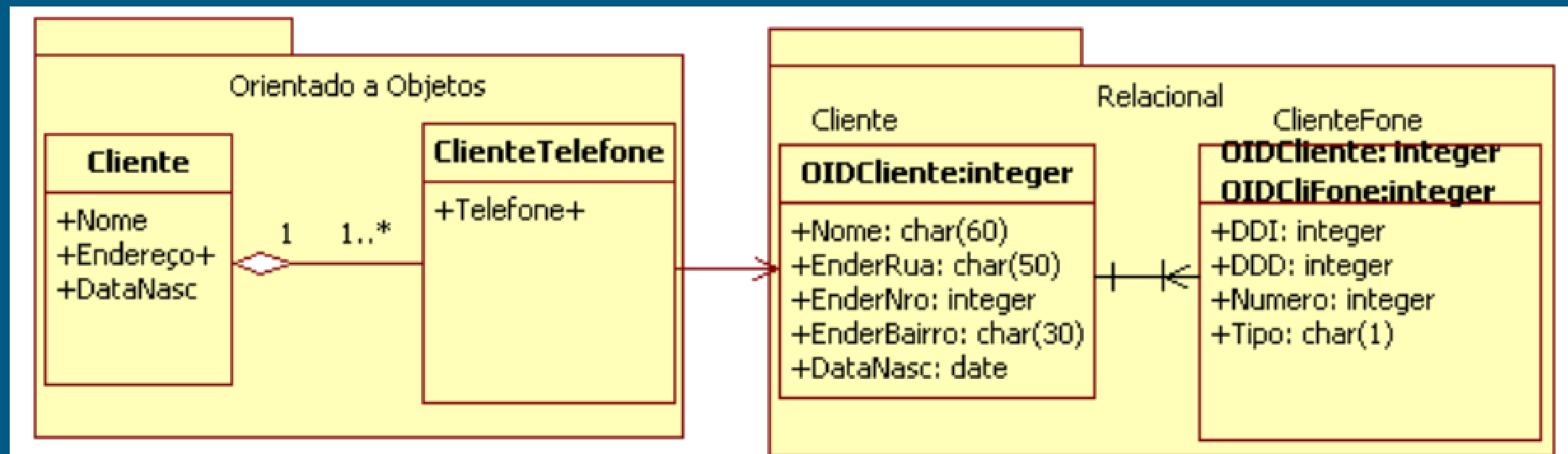
<https://consultabd.wordpress.com/2019/08/21/modelo-relacional/>



fonte: <https://prosimples.com/conceitos-basicos-de-banco-de-dados/>

Evolução dos Modelos

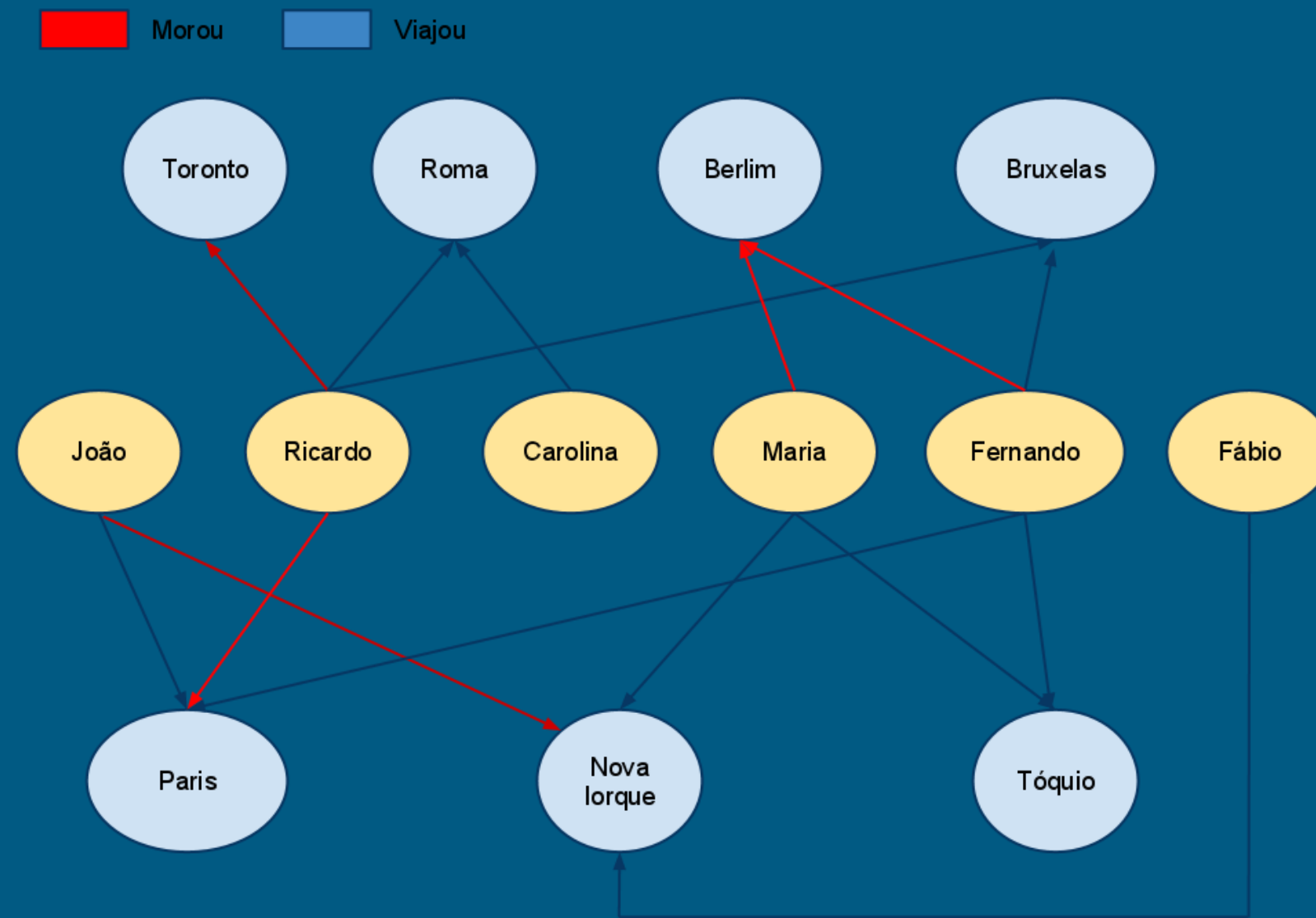
Orientado a Objetos – Integração com programação orientada a objetos



fonte: <https://www.devmedia.com.br/tecnicas-de-mapeamento-objeto-relacional-revista-sql-magazine-40/6980>

Evolução dos Modelos

NoSQL – Flexibilidade, escalabilidade e diversos formatos (documentos, chave-valor, grafos)



Referências

<https://castelano.com.br/site/aulas/bd/Aula%2003%20-%20Modelos%20de%20Banco%20de%20Dados.pdf>

<https://consultabd.wordpress.com/2019/08/21/modelo-relacional/>

<https://prosimples.com/conceitos-basicos-de-banco-de-dados/>

https://www.cadcobol.com.br/db2_novo_conceito_banco_de_dados.htm

<https://www.devmedia.com.br/tecnicas-de-mapeamento-objeto-relacional-revista-sql-magazine-40/6980>

<https://www.devmedia.com.br/banco-de-dados-nosql-um-novo-paradigma-revista-sql-magazine-102/25918>