



MODELAGEM DE DADOS

Aula 2 – Classificação de Bancos de Dados

Curso de Ciência da Computação

Dr. Rodrigo Xavier de Almeida Leão

Cientista de Dados



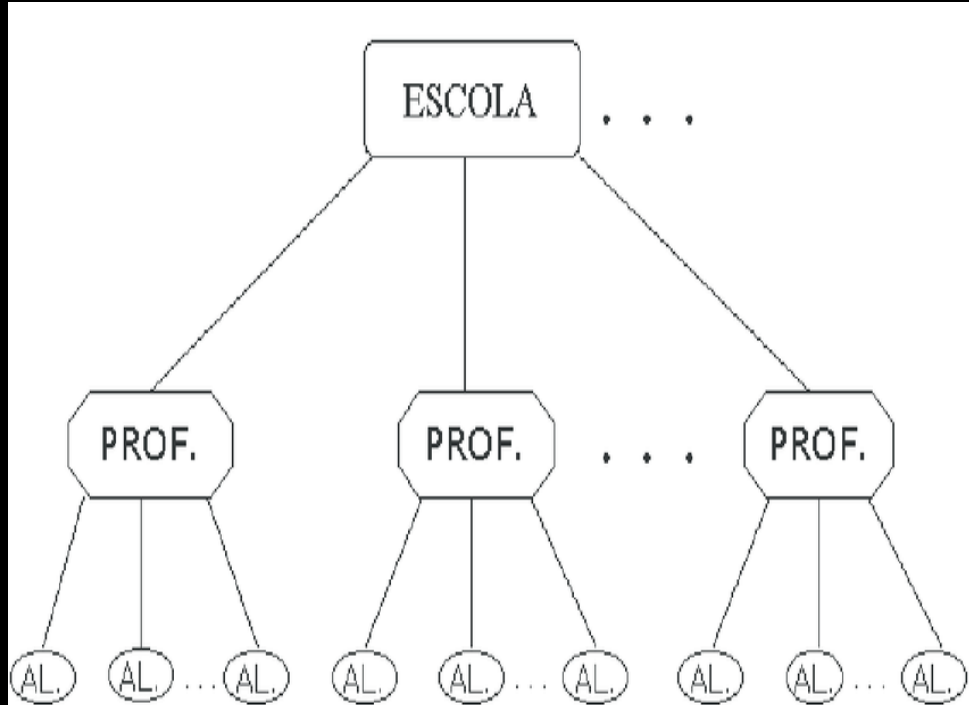
CLASSIFICAÇÃO DE BANCOS DE DADOS

- As primeiras aplicações de banco de dados mantinham as informações de grandes empresas, como multinacionais, hospitais, universidades e bancos. Grande parte dos sistemas que utilizava os bancos de dados desta época, usava computadores de grande porte: os mainframes.
- Esses computadores eram muito caros e possuíam somente uma interface para a linguagem de programação, e cada sistema desenvolvido levava muito tempo para ser programado, pois todas as transações realizadas no banco de dados eram programadas, testadas e depuradas.

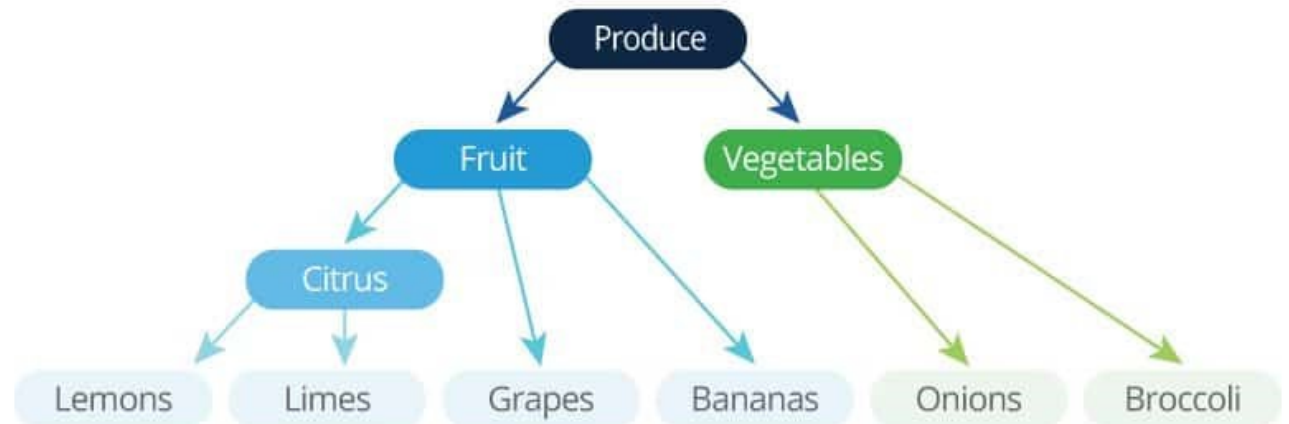
Hierárquicos

- Os dados são organizados em uma estrutura hierárquica semelhante a uma árvore, com registros pai e filhos. Cada registro pai pode ter vários registros filhos, mas cada registro filho só pode ter um pai.
- Caso fosse necessária a adição de uma nova informação (tabela ou campo), o banco de dados em sua totalidade precisaria ser reorganizado ou redefinido.
- Foi popular nas décadas de 1960 e 1970

Hierárquicos



Hierarchical Database Model



The hierarchical database model has parent-child relationships that are one-to-one or one-to-many.

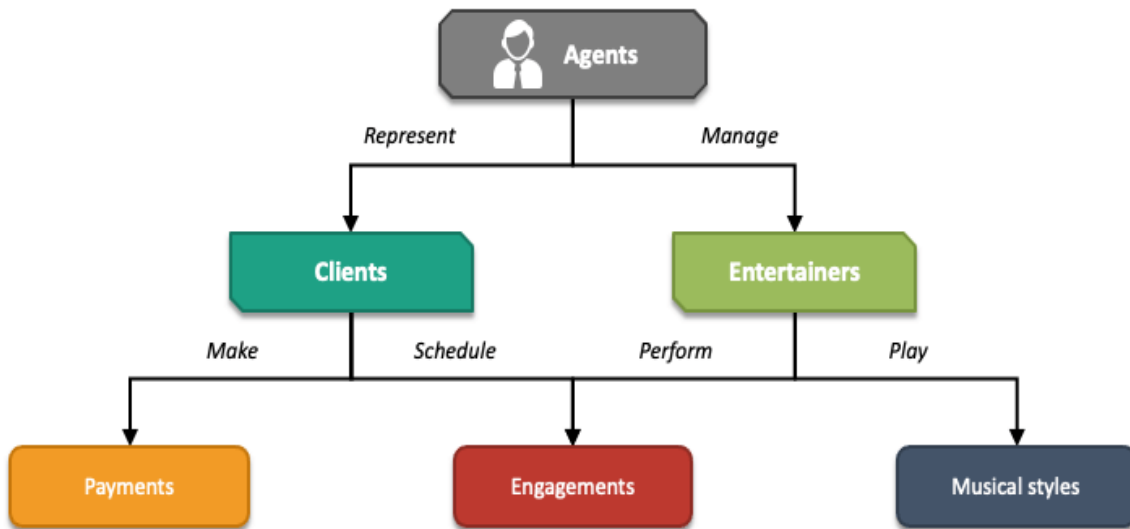
REDE

- É uma extensão do modelo hierárquico, permitindo que um registro filho tenha mais de um pai.
- Isso é alcançado através do uso de estruturas de dados chamadas conjuntos de registros (sets).
- O modelo de rede foi desenvolvido para superar algumas limitações do modelo hierárquico, mas acabou sendo substituído pelo modelo relacional.

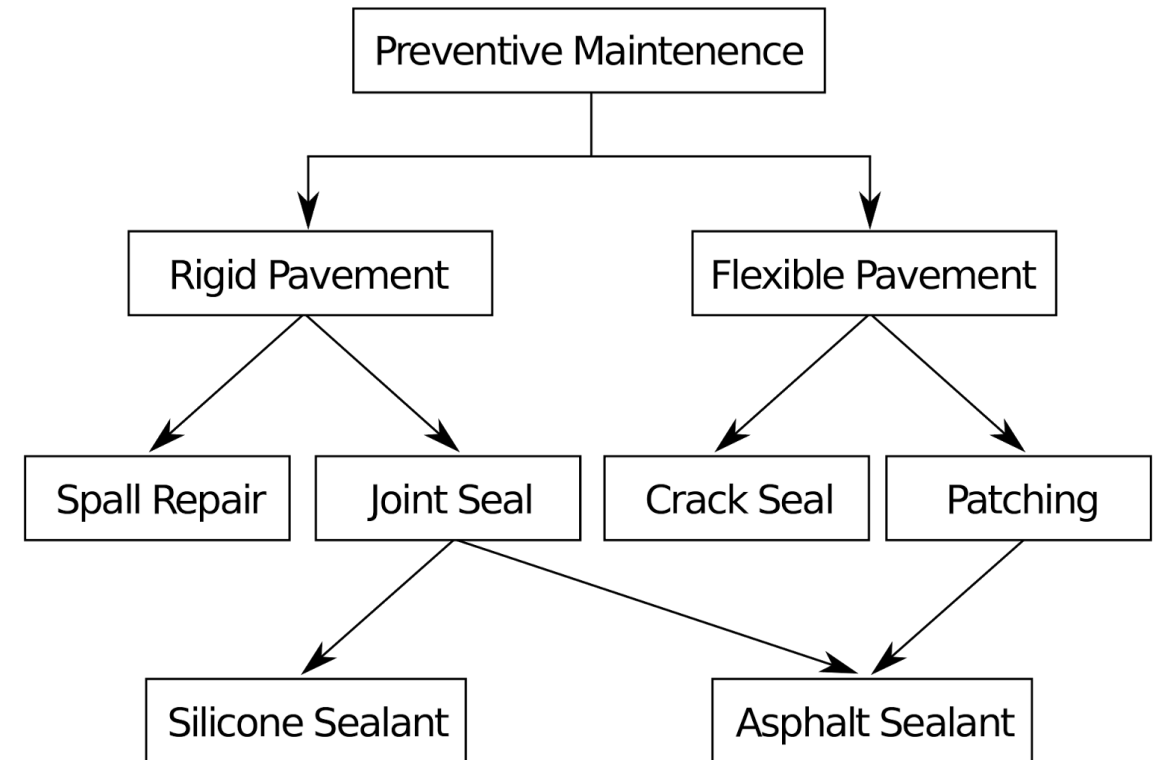
REDE

NETWORK DATABASE MODEL

Enter your sub headline here



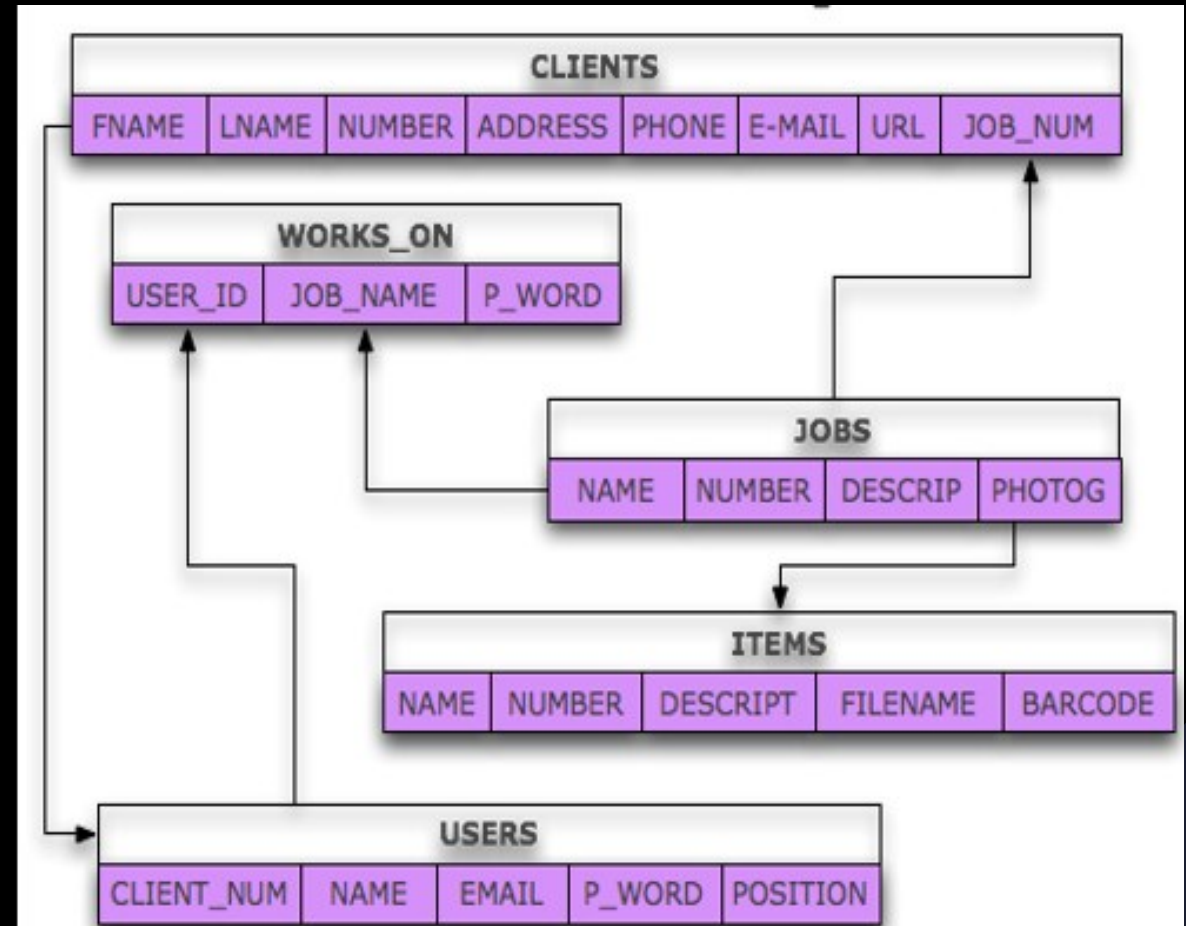
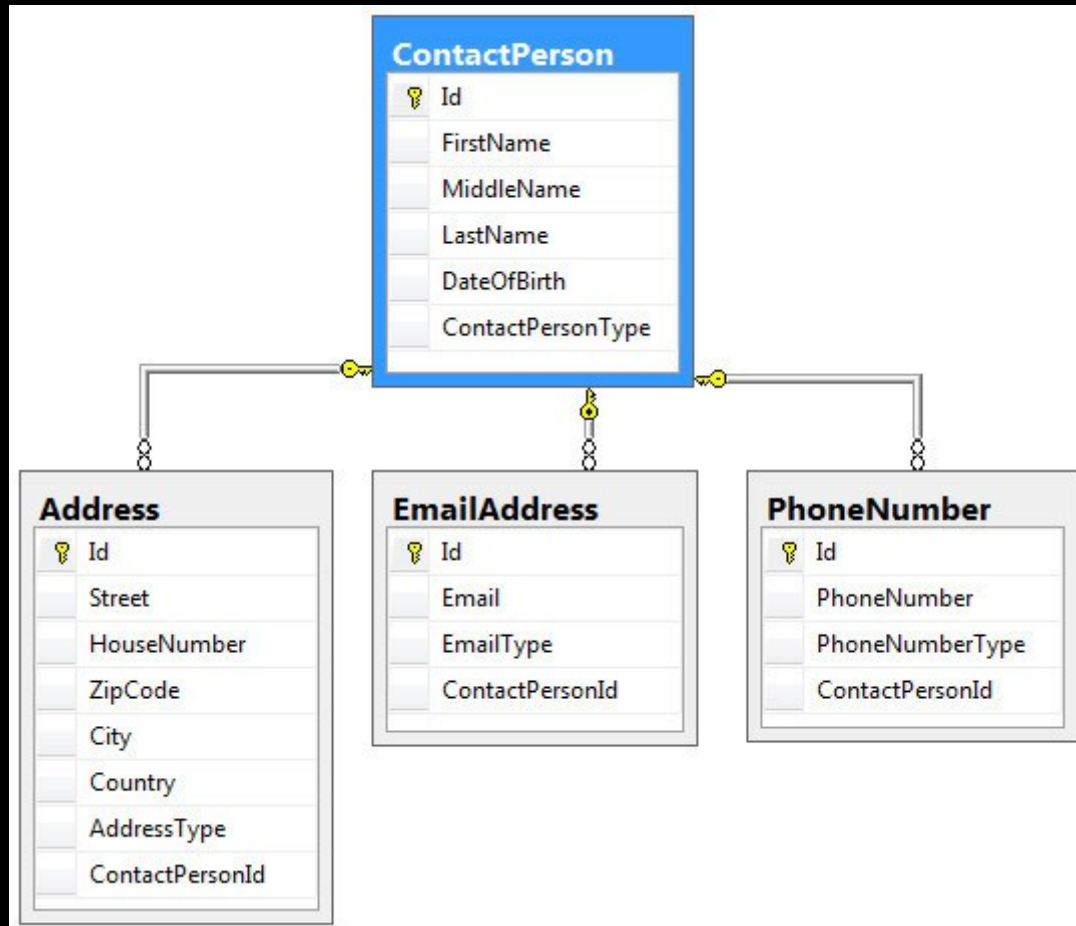
Network Model



RELACIONAL

- Os bancos de dados relacionais começaram a surgir comercialmente a partir de 1980.
- Os dados são organizados em tabelas com linhas e colunas.
- Cada tabela representa uma entidade e cada linha representa uma instância dessa entidade.
- As relações entre as entidades são representadas por meio de chaves estrangeiras.

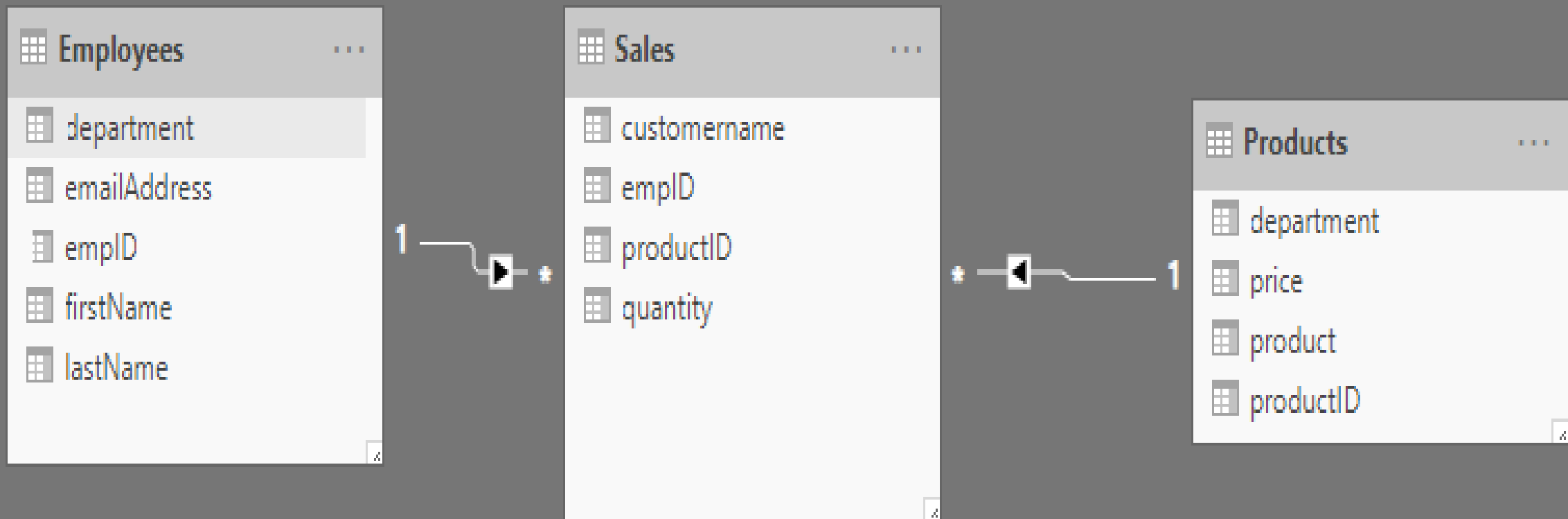
RELACIONAL



employee	empID	department	product	qty	price	orderAmount
Kelli Hinojos	1	Game	Settlers of Air	1	24.99	24.99
Jeffrey Reiss	5	Sports	Driver - Stiff Shaft	1	399.99	399.99
Roselyn James	7	Clothing	V-Neck T-Shirt	1	19.99	19.99
Lavonna Domingo	5	Sports	Golf Balls - Dual Core	3	32.5	97.5

employee	empID	productID	quantity
Kelli Hinojos	1	82	1
Jeffrey Reiss	5	24	1
Roselyn James	7	67	1
Lavonna Domingo	5	42	3
Hermina Leslie	7	88	4
Jess Dammann	6	19	1

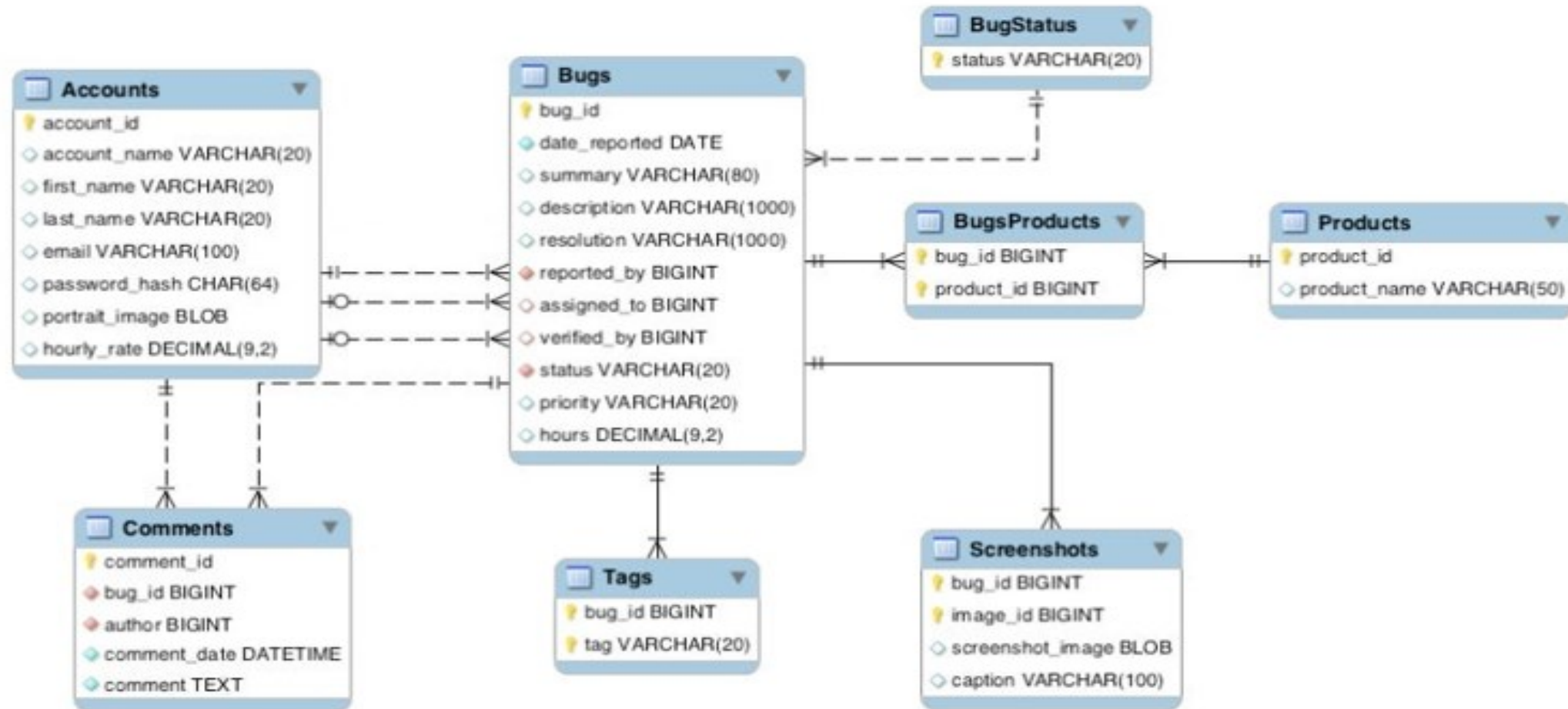
productID	department	product	price
3	Sports	Frisbee Golf Set	98
4	Sports	Weighted Bands	8.99
8	Automotive	Window Scrape	9.99
16	Clothing	Bathing Trunks	42.99
19	Automotive	Tire Guard	44.99



ORIENTADO A OBJETO

- Neste modelo, os dados são representados como objetos, que podem conter dados, bem como métodos para manipular esses dados.
- Este modelo é mais complexo do que o modelo relacional e é adequado para situações em que a estrutura dos dados é altamente complexa e variável, como em sistemas de engenharia, modelagem de simulação e desenvolvimento de software orientado a objetos

ORIENTADO A OBJETO

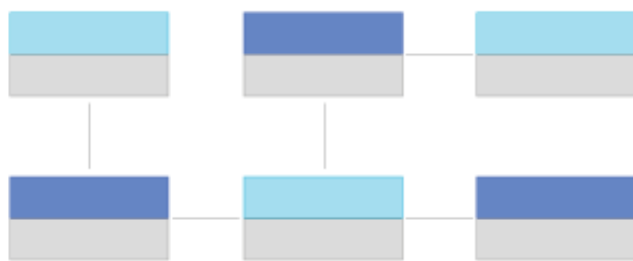


NoSQL

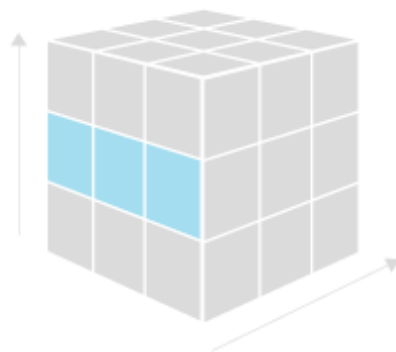
- Os bancos de dados NoSQL são projetados para lidar com grandes volumes de dados e oferecem maior flexibilidade de esquema em comparação com os bancos de dados relacionais.
- Existem vários tipos de bancos de dados NoSQL, incluindo bancos de dados de documentos (como MongoDB), bancos de dados de chave-valor (como Redis), bancos de dados de colunas (como Cassandra) e bancos de dados de grafos (como Neo4j).

SQL

Relational

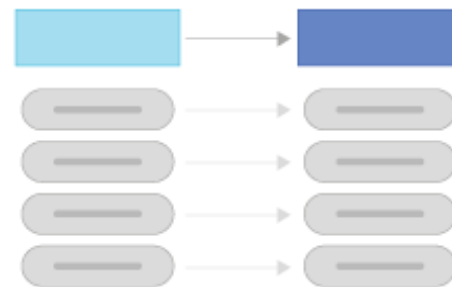


Analytical



NoSQL

Key - Value



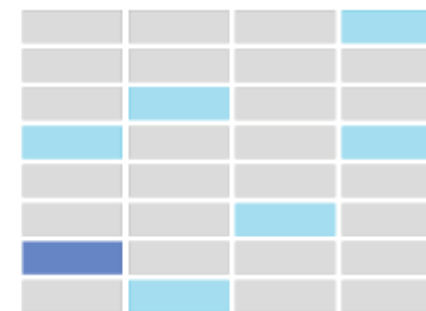
Graph



Document



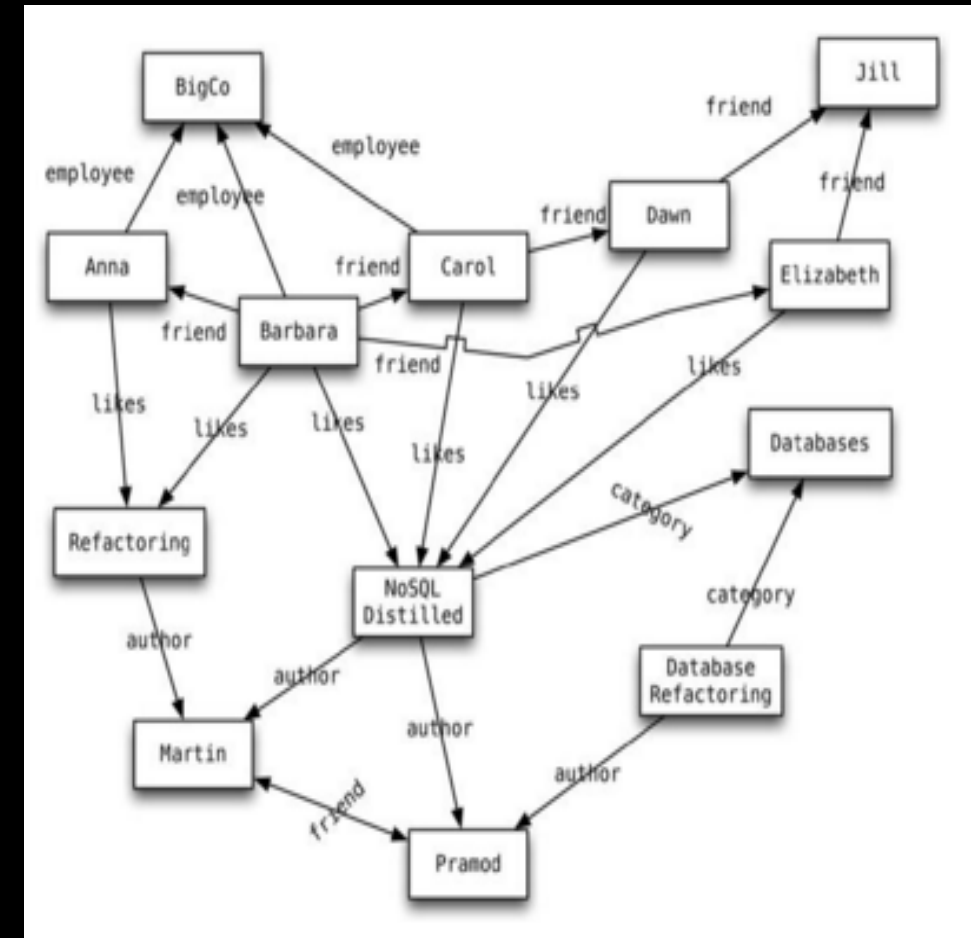
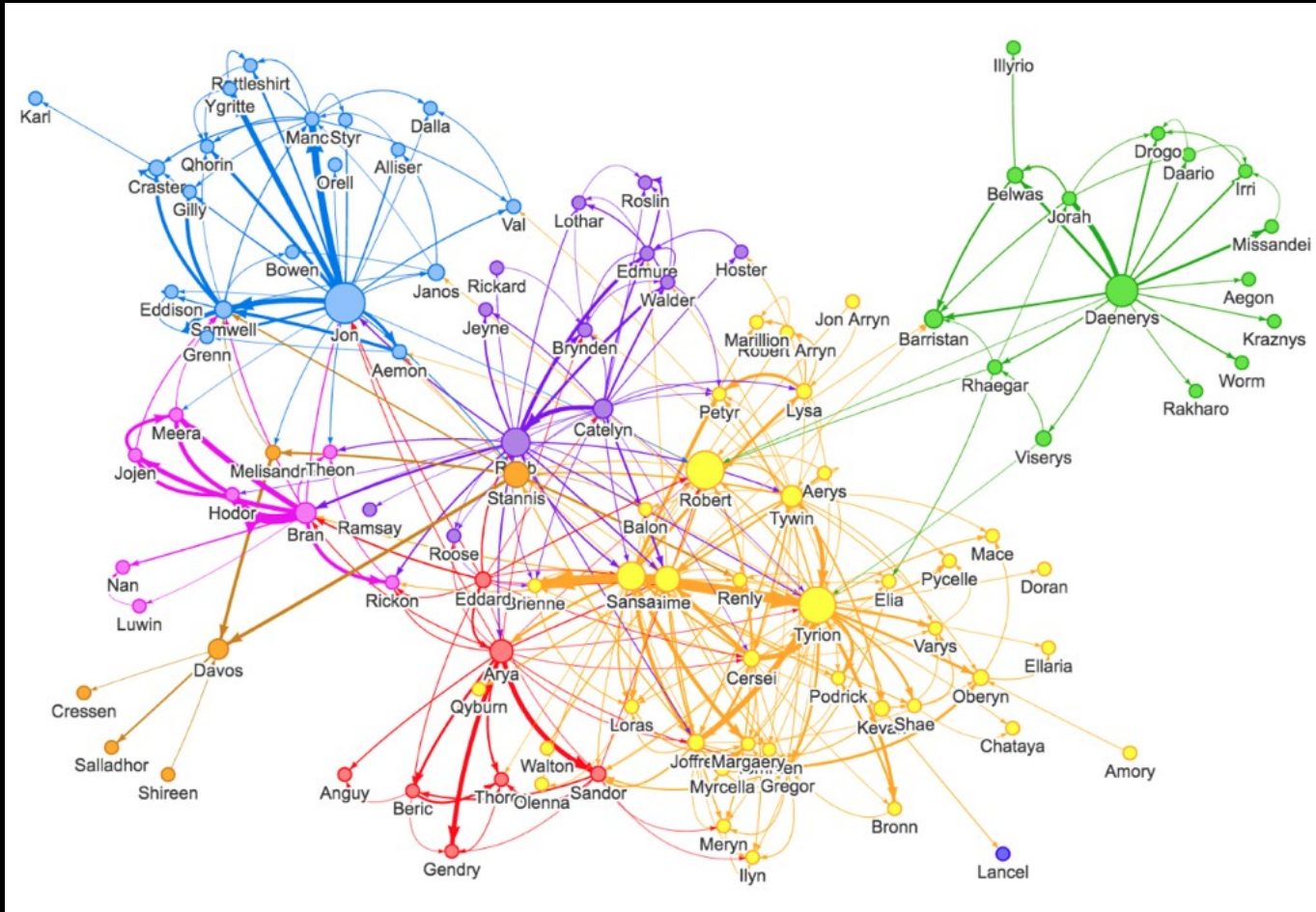
Wide Column



Banco de Dados de Grafos

- Os bancos de dados de grafos são otimizados para armazenar e consultar dados relacionados por meio de grafos. Eles são úteis para representar redes complexas de dados, como redes sociais ou sistemas de recomendação.
- **Exemplos incluem** Neo4j, Amazon Neptune e Microsoft Azure Cosmos DB.

Banco de Dados de Grafos



XML

- O XML (Extensible Markup Language) é uma linguagem de marcação que permite a representação de dados de forma estruturada e legível por máquina.
- Ele é frequentemente utilizado para a troca de dados entre sistemas diferentes devido à sua flexibilidade e capacidade de representar uma ampla variedade de tipos de dados.

XML

- **Estrutura Hierárquica:** O XML organiza os dados em uma estrutura hierárquica de tags aninhadas, o que facilita a representação de dados complexos e relacionados de forma organizada. Essa estrutura hierárquica pode ser facilmente mapeada para a estrutura de dados de diferentes sistemas de banco de dados.
- **Independência de Plataforma:** O XML é independente de plataforma e linguagem de programação, o que significa que pode ser facilmente lido e processado por diferentes sistemas.

XML

- **Flexibilidade:** O XML é altamente flexível e extensível, permitindo a definição de esquemas personalizados para representar tipos de dados específicos. Isso permite que diferentes sistemas definam suas próprias estruturas de dados e regras de validação, enquanto ainda podem trocar dados entre si usando XML.

XML

- **Interoperabilidade:** Como o XML é amplamente suportado por uma variedade de tecnologias e ferramentas, ele facilita a interoperabilidade entre sistemas heterogêneos. Os sistemas podem trocar dados em formato XML e usar tecnologias como parsers XML para processar esses dados.
- **Padrões de Intercâmbio de Dados:** Existem muitos padrões e especificações que utilizam XML para facilitar a troca de dados entre sistemas, tais como SOAP (Simple Object Access Protocol) para serviços web, XML-RPC (XML Remote Procedure Call), e outros.

XML

<book>

<title> Harry Potter and the Philosopher's Stone </title>

<author> J.K. Rowling </author>

<year> 1997 </year>

</book>

JSON

- O JSON (JavaScript Object Notation) é um formato de intercâmbio de dados leve, baseado em texto, que é amplamente utilizado para transmitir dados estruturados entre um servidor e um cliente, e entre diferentes sistemas.
- Ele foi inspirado na sintaxe de objetos literais do JavaScript, mas é independente de linguagem e pode ser facilmente interpretado e gerado por uma variedade de linguagens de programação

JSON

- O JSON utiliza uma sintaxe simples e legível, composta por pares chave-valor, separados por dois pontos (:), e separados por vírgulas (,).
- Os valores podem ser strings, números, booleanos, arrays, objetos ou null.
{
 "book": {
 "title": "Harry Potter and the Philosopher's Stone",
 "author": "J.K. Rowling",
 "year": 1997
 }
}

XML vs JSON

- No exemplo XML, cada campo (como título, autor e ano) é envolvido por tags de abertura e fechamento (<title>, </title>, <author>, </author>, <year>, </year>), o que aumenta a quantidade de caracteres necessários para representar os dados.
- Por outro lado, no exemplo JSON, os dados são representados de forma mais concisa usando uma estrutura de pares chave-valor dentro de objetos e arrays.

XML Verbose

- O **XML é mais verbose** em comparação com o JSON, ou seja, o XML tende a exigir mais caracteres para representar a mesma informação em comparação com o JSON.
- Isso ocorre devido à natureza da sintaxe do XML, que requer tags de abertura e fechamento para cada elemento, enquanto o JSON usa uma estrutura de pares chave-valor mais compacta

SGBD's não Freeware

- **Oracle Database:** Desenvolvido pela Oracle Corporation, o Oracle Database é um dos SGBDs mais poderosos e amplamente utilizados em empresas. Ele oferece uma ampla gama de recursos para gerenciamento de dados em grande escala, alta disponibilidade e segurança avançada. O Oracle Database é licenciado comercialmente e requer uma taxa de licenciamento para uso em ambientes de produção.

SGBD's não Freeware

- **Microsoft SQL Server:** Desenvolvido pela Microsoft, o SQL Server é um SGBD relacional amplamente utilizado em ambientes corporativos. Ele oferece recursos avançados de gerenciamento, segurança e análise de dados. O SQL Server está disponível em várias edições, incluindo uma edição gratuita (SQL Server Express), mas as edições comerciais requerem uma licença paga.

SGBD's não Freeware

- **IBM Db2:** O IBM Db2 é um SGBD relacional desenvolvido pela IBM, oferecendo suporte a ambientes de mainframe, servidor e nuvem. Ele fornece recursos avançados de gerenciamento de dados, escalabilidade e desempenho. O Db2 está disponível em várias edições, algumas das quais requerem uma licença paga.

SGBD's não Freeware

- **SAP HANA:** Desenvolvido pela SAP, o SAP HANA é um SGBD em memória que oferece recursos avançados para análise em tempo real e processamento de dados de alta velocidade. É amplamente utilizado em empresas para suportar aplicativos empresariais críticos. O SAP HANA geralmente requer uma licença paga.

SGBD's não Freeware

- **SAP HANA:** Desenvolvido pela SAP, o SAP HANA é um SGBD em memória que oferece recursos avançados para análise em tempo real e processamento de dados de alta velocidade. É amplamente utilizado em empresas para suportar aplicativos empresariais críticos. O SAP HANA geralmente requer uma licença paga.

SGBD's Freeware

- **MySQL:** É um dos SGBDs mais populares do mundo, conhecido por sua confiabilidade, desempenho e facilidade de uso. O MySQL é distribuído sob a licença GNU GPL (General Public License) e está disponível gratuitamente para uso.
 - **PostgreSQL:** É um poderoso SGBD relacional de código aberto que oferece recursos avançados, como suporte a transações ACID, integridade referencial e extensibilidade. O PostgreSQL é distribuído sob uma licença de software livre e está disponível gratuitamente para uso.
-

SGBD's Freeware

- **SQLite:** É um SGBD embutido de código aberto que não requer um servidor separado, tornando-o ideal para aplicativos móveis e pequenos projetos. O SQLite é de domínio público e pode ser usado gratuitamente para qualquer propósito, comercial ou não comercial.
- **MariaDB:** É um fork do MySQL desenvolvido pela comunidade após a aquisição do MySQL pela Oracle Corporation. O MariaDB é distribuído sob a licença GPL e é uma alternativa de código aberto ao MySQL.

SGBD's Freeware

- **Firebird:** É um SGBD relacional de código aberto que oferece suporte a transações ACID e é adequado para aplicativos embarcados, de desktop e de servidor. O Firebird é distribuído sob a licença de Interbase Public License (IPL) e pode ser usado gratuitamente.

Principais Características dos SGBD's

- Permite inclusão, exclusão, seleção, ordenação e junção de registros de entidades.
- Possibilita a cópia e a exclusão de entidades.
- Estabelece relações entre as entidades e a criação de chaves.
- Permite a importação ou exportação de dados entre outras bases de dados.
- Possibilita a alteração da estrutura de campos e entidades.
- Permite consultas e relatórios da base de dados.
- Possibilita a criação de usuários com permissão de acesso individualizados.