# Introdução aos BD não Relacionais

-1970 – BD Relacional

-1998 – NoSQL – Carlos Strosi

-2009 – Reaparição do NoSQL

NoSQL(Not Only SQL)

Diferenças entre Relacionais e não Relacionais.

Texto

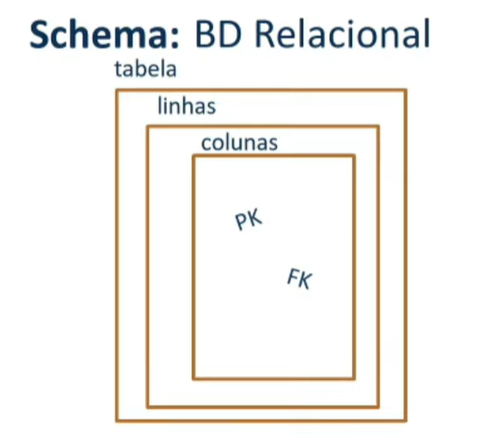
Descrição gerada automaticamente com confiança baixa

Texto

Descrição gerada automaticamente com confiança baixa

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente



Uma imagem contendo Texto

Descrição gerada automaticamente

Disco rígido de computador

Descrição gerada automaticamente com confiança média

Ícone

Descrição gerada automaticamente

Texto

Descrição gerada automaticamente

Vantagens dos BD NoSQL

* Flexibilidade
* Escalabilidade
* Alta-Performance

Interface gráfica do usuário

Descrição gerada automaticamente

## SGBD Orientado a Grafos (Graph)

Gráfico de bolhas

Descrição gerada automaticamente com confiança baixa

Comum em detecção de fraudes, mecanismos de recomendação, redes sociais, sistemas de arquivos, games...

Os nós seria o dado nesse modelo grafo.

Neo4j – Linguagem utilizada é a Cypher

## SGBD – Coluna/Família de Colunas

Interface gráfica do usuário

Descrição gerada automaticamente

Interface gráfica do usuário, Texto

Descrição gerada automaticamente

Registro de transações: compras, resultados de testes, filmes assistidos e localização mais recente do filme.

Rastreando praticamente qualquer coisa, incluindo status do pedido pacotes etc.

Linguagem do Cassada – CQL

## SGDB – Baseados em Chaves-Valor

Interface gráfica do usuário, Diagrama

Descrição gerada automaticamente

Armazena um conjunto de dados, seja ele simples ou complexo, identificados por um identificador exclusivo.

+ Bom desempenho em aplicações na nuvem.

- Menor capacidade de busca.

Uso: Cache, Sessão de usuário, carrinhos de compras

## SGBD – Relacionado a Documentos

Dados e Documentos autocontidos e auto descritivos.

Permite redundância e inconsistência.

Livre de esquemas podendo utilizar JSON, XML entre outros.

MongoDB.

* Código Aberto
* Alta Performance
* Schema-free
* Utiliza Json para armazenamento dos dados
* Suporte a índices
* Auto-Sharding
* Map-Reduce
* GridFS

(Para criar essa seta(🡺) só colocar Igual, Igual e sinal > (==>))

Document 🡺 Tupla/Registro

Collection 🡺 Tabela

Embedding/linking 🡺 Join

Quando usar:

* Grande Volume de dados.
* Dados não necessariamente estruturados.

Quando não utilizar:

* Necessidade de relacionamentos/Joins.
* Propriedade ACID e transações são importantes.
* Curiosidade: Diversas entidades de pagamento não homologam sistemas cujos dados financeiros dos clientes não estejam em BD Relacionais tradicionais.

# Schema Design Embedding vs Referência

Documentos autocontido

Documentos com depenica de outros documentos ou collections



Relacionamento por Embedding:

Pros:

Consulta informações em uma única query

Atualiza o registro em uma única operação

Contras :

Limite de 16 MB por documento

Relacionamento por referencia:

Pros:

Documentos pequenos

Não duplica informações

Usado quando os dados não são acessados em todas as consultas

Contras:

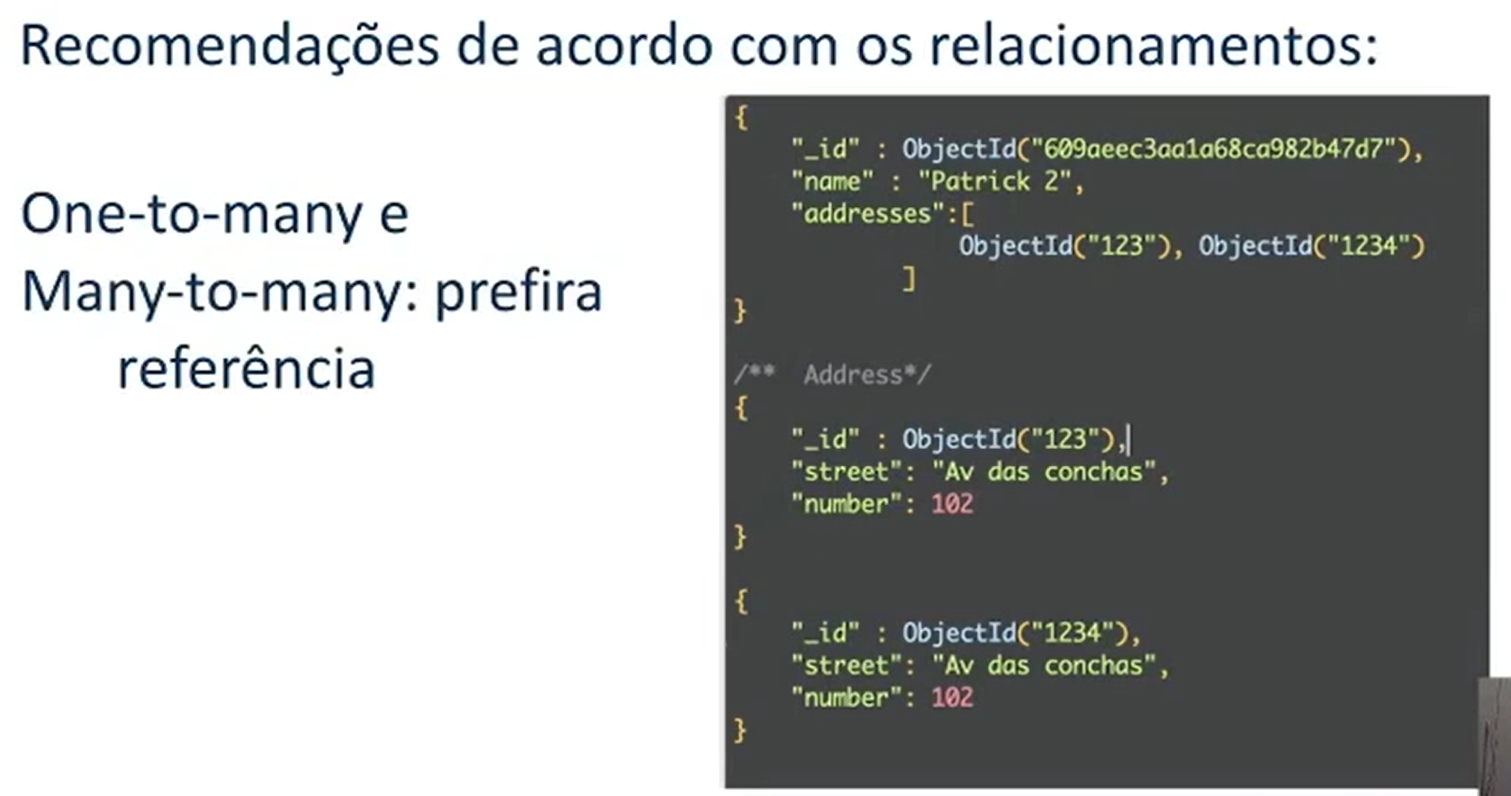
Duas ou mais queries ou utilização do $lookup

Texto

Descrição gerada automaticamente

Texto

Descrição gerada automaticamente



Boas Práticas

Evite documentos muito grandes

Use nome campos objetivos e curtos

Analise as suas queries utilizando explain()

Atualize apenas os campos alterados

Evite negações em queries

Listas/Arrays dentro dos documentos não podem crescer sem limite

JSON E BSON

BSON:

É uma serialização codificada em binários de documentos semelhantes a JSON

Contém extensões que permitem a representação de tipos de dados que não fazem parte da especialização JSON. Por exemplo, BSON tem um tipo Date, ObjetctID

Mongo utiliza o Bson.

## Links Uteis

<https://www.mongodb.com/blog/post/building-with-patterns-a-summary>

<https://www.mongodb.com/docs/manual/aggregation/>