O Engenheiro de Dados é o profissional responsável por gerenciar, otimizar, supervisionar e monitorar a recuperação, armazenamento e distribuição de dados em toda uma organização.

Realiza as criações e conservações de pipelines que transformam os dados brutos que estão nos mais variados formatos, desde bancos de dados transacionais até arquivos de texto, em um formato que permita ao Cientista de Dados começar seu trabalho. Com o nível de segurança exigido pela empresa

Um Engenheiro de Dados precisa ser bom em:

- Arquitetar sistemas distribuídos
- Criar pipelines confiáveis
- Combinar fontes de dados
- Criar a arquitetura de soluções
- Colaborar com a equipe de Data Science e construir as soluções certas para essas equipes

No ambiente de suas competências esse profissional tem a incumbência de assimilar o papel dos Bancos de Dados Relacionais (SQL) e Não Relacionais (NoSQL). Pois atualmente, está acontecendo uma grande produção na quantidade de dados. Com a internet das coisas, temos uma rede de dispositivos capazes de coletar, transmitir e processar dados e com essa grande quantidade de dados, o armazenamento e processamento destes é o novo desafio.

As grandes corporações vêm utilizando os bancos de dados relacionais, entretanto, a produção elevada de dados atribui, a esses bancos, dificuldades de escalabilidade e performance, considerando o respeito as propriedades ACID e as formas normais. Atualmente já é trabalhado com dados, muitas vezes sem estrutura fixa, o que acaba no uso de SGBDs não relacionais, também conhecidos como SGDBs NoSQL.

Em banco de dados relacional a arquitetura mais difundida na literatura é a Arquitetura “Three-Schema” (também conhecida como arquitetura ANSI/SPARC), proposta por Tsichritzis & Klug em 1978.

A meta desta arquitetura, é separar as aplicações de usuários da base de dados física. Nesta arquitetura, esquemas podem ser definidos em três níveis:

- O nível interno tem um esquema que descreve a estrutura de armazenamento físico da base de dados. O esquema interno usa um modelo de dados físico e descreve todos os detalhes de armazenamento de dados e caminhos de acesso à base de dados;
- O nível conceitual tem um esquema que descreve a estrutura de toda a base de dados. O esquema conceitual é uma descrição global da base de dados, que omite detalhes da estrutura de armazenamento físico e se concentra na descrição de entidades, tipos de dados, relacionamentos e restrições. Um modelo de dados de alto-nível ou um modelo de dados de implementação podem ser utilizados neste nível;

- O nível externo ou visão possui esquemas externos ou visões de usuários. Cada esquema externo descreve a visão da base de dados de um grupo de usuários da base de dados. Cada visão descreve, tipicamente, a parte da base de dados que um particular grupo de usuários está interessado e esconde deste o restante da base de dados. Um modelo de dados de alto-nível ou um modelo de dados de implementação podem ser usados neste nível.

A arquitetura “three-schema” pode ser utilizada para explicar conceitos de independência de dados, que podem ser definidos como a capacidade de alterar o esquema de um nível sem ter que alterar o esquema no próximo nível superior. Dois tipos de independência de dados podem ser definidos:

- Independência lógica de dados: É a capacidade de alterar o esquema conceitual sem ter que mudar os esquemas externos ou programas de aplicação. Pode-se mudar o esquema conceitual para expandir a base de dados, com a adição de novos tipos de registros (ou itens de dados), ou reduzir a base de dados removendo um tipo de registro. Neste último caso, esquemas externos que se referem apenas aos dados remanescentes não devem ser afetados.

- Independência física de dados: é a capacidade de alterar o esquema conceitual externo. Mudanças no esquema interno podem ser necessárias devido a alguma reorganização de arquivos físicos para melhorar o desempenho nas recuperações e/ou modificações. Após a reorganização, se nenhum dado foi adicionado ou perdido, não haverá necessidade de modificar o esquema conceitual.

Quando o tratamos de armazenamento de dados é para Big Data tem que levar em consideração as seguintes características:

- O tamanho das coleções de dados;

- O conteúdo das coleções, que tendem a variar de texto, BLOBS e conteúdo estruturado e semi-estruturado;

- O suporte à persistência, à medida que as coleções crescem, suportes de armazenamento mais sofisticados são necessários com o desafio de decidir como organizar todos esses dados em um conjunto independente de discos (RAID) ou suportar o armazenamento dos mesmos em nuvem;

- O acesso a dados, pequenas coleções requerem técnicas de acesso a dados diferentes das grandes coleções de dados que estão distribuídas em diferentes meios de armazenamento.

Conforme com a organização NoSQL DB (http://nosql-database.org), a forma como NoSQL armazena os dados será a próxima geração de bancos de dados: não-relacionais, distribuídos, horizontalmente escaláveis e com alta performance.

Existem hoje quatro tipos de bancos de dados NoSQL existentes. São eles:

Key-value**:** Sistemas que armazenam valores e um índice para encontrá-los e que é baseado em um ID (key).

Quando usar

- Desde que o acesso ao banco de dados é apenas através de uma chave primária, geralmente o desempenho é grande e o dimensionamento é fácil via sharding.
- Armazenamento de informações por sessão (cache e distribuído)
- Perfis de usuário / preferências
- Dados do carrinho de compras (cluster distribuído)

Quando não usar

- Uma vez que você só pode consultar pela chave primária, isso significa que você não pode ver a estrutura dentro de uma agregação.
- Quando houver relacionamentos entre os dados
- Transações multi-task
- Consulta por dados
- Operações por conjuntos

Documento**:** Sistemas que armazenam documentos providenciando índices e mecanismos simples de queries.

Quando usar

- Registro de eventos
- Sistemas de gerenciamento de conteúdo, plataformas de blogs
- Web Analytics ou Analytics em Tempo Real
- Aplicações de comércio eletrônico

Quando não usar

- Transações complexas abrangendo diferentes operações
- Consultas estrutura de diferentes agregações

Colunar**:** Sistemas que armazenam extensões de registros e que podem ser particionados verticalmente e horizontalmente através de nós.

Quando usar

- Registro de eventos
- Sistemas de gerenciamento de conteúdo, plataformas de blogs
- Contadores
- Tempo de uso (expiração)

Quando não usar

- Sistemas que exigem transações ACID para leitura e gravação

- Primeiros protótipos ou picos de tecnologia iniciais, o custo de uma alteração de consulta pode ser maior em comparação com uma alteração de esquema (problema que o Netflix teve quando iniciou)

Grafos**:** Sistemas que armazenam modelos de dados como grafos onde nós podemos representar conteúdo modelado como document ou key-value estruturado e arcos que representam a relação entre o dado modelado pelo nó.

Quando usar

- Dados conectados
- Serviços de Roteamento, Despacho e Localização
- Motores de Recomendação

Quando não usar

- Quando você deseja atualizar todo o conjunto ou um subconjunto de entidades

Para concluir, no dia a dia do trabalho do Engenheiro de Dados deve analisar a descrição do trabalho envolvido, para escolher qual paradigma é necessário para entregar o produto. As ferramentas que estão disponíveis, servem somente para facilitar o processo de trabalho. O que mais importa é o domínio na base de conhecimento das tecnologias envolvidas.