



Olá, aluno(a)!
Seja bem-vindo(a) à aula interativa!

Você entrará na reunião com a câmera e o microfone desligados.

Sua presença será computada através da enquete.
Fique atento(a) e não deixe de respondê-la!

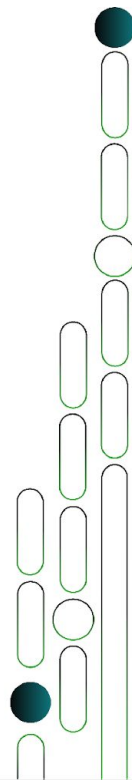


Aprenda com quem faz

> Bancos de Dados NoSQL

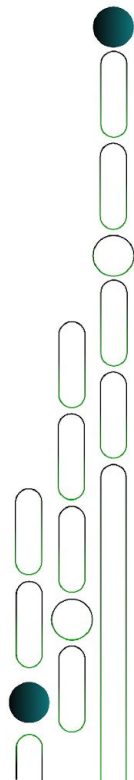
Segunda Aula Interativa

Prof. Ricardo Brito Alves



Nesta aula

- Desafio.
- Tópicos da disciplina e temas interessantes.



MongoDB Compass



MongoDB Compass - cluster0.vo6xw.mongodb.net:27017

Connect View Help

Local

3 DBS 7 COLLECTIONS

☆ FAVORITE

HOSTS

- cluster0-shard-00-01.vo6xw...
- cluster0-shard-00-02.vo6xw...
- cluster0-shard-00-00.vo6xw...

CLUSTER

Replica Set (atlas-13e3rw-...)

3 Nodes

EDITION

MongoDB 4.2.10 Enterprise

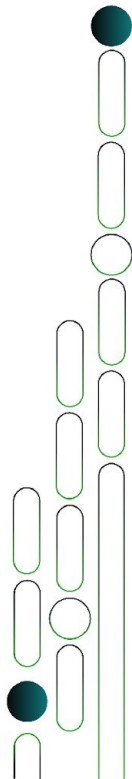
Filter your data

- > admin
- > config
- > local

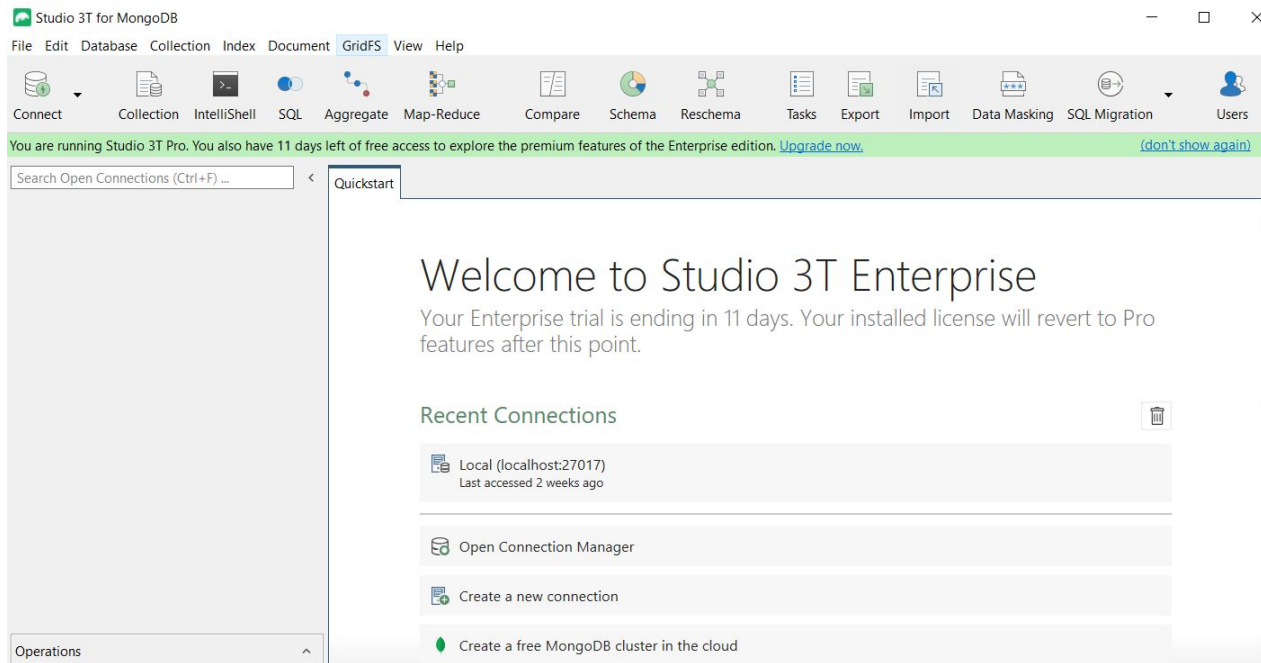
Databases Performance

CREATE DATABASE

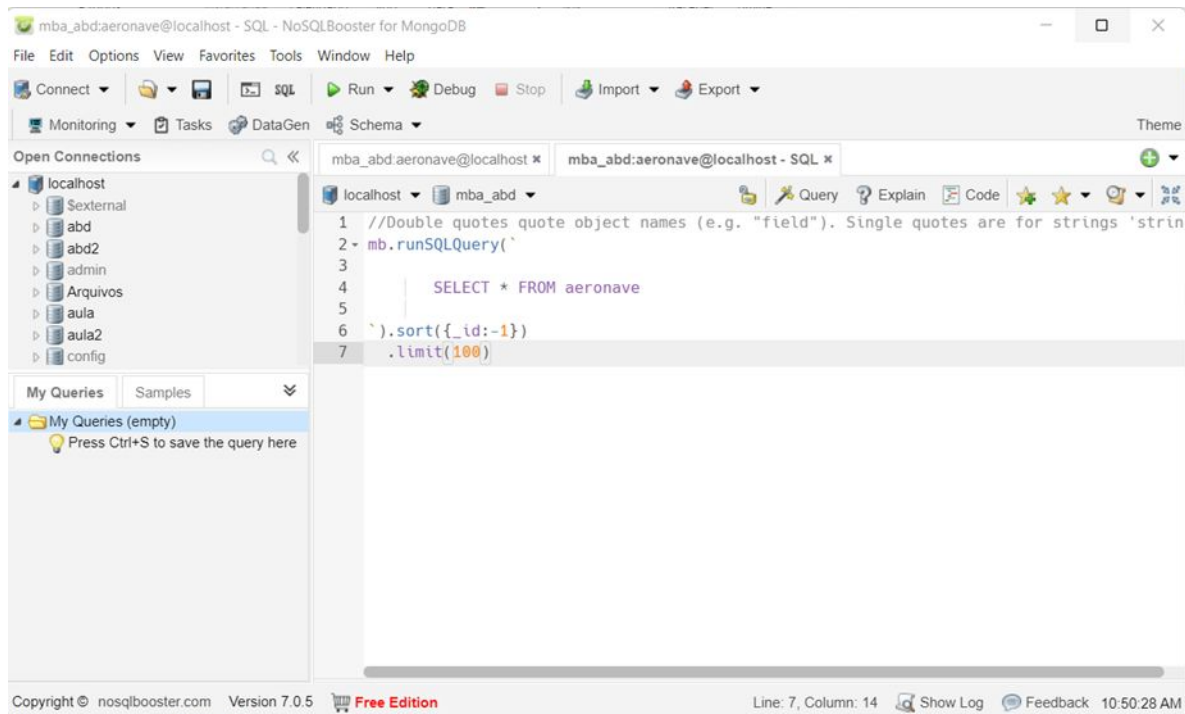
Database Name ^	Storage Size	Collections	Indexes
admin	0.0B	0	0
config	0.0B	1	0
local	0.0B	6	0



Studio 3T for MongoDB

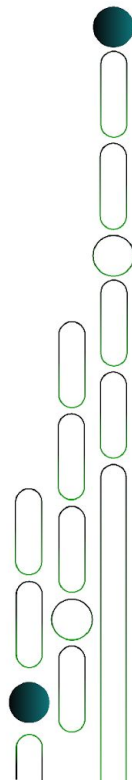


NoSQLBooster for MongoDB

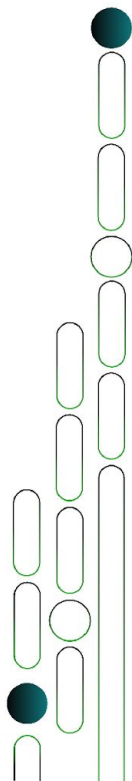
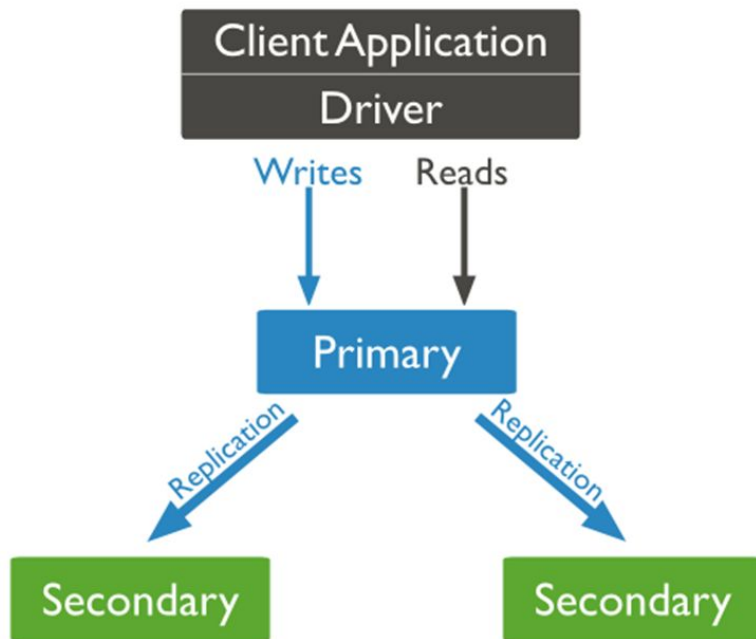


DBA

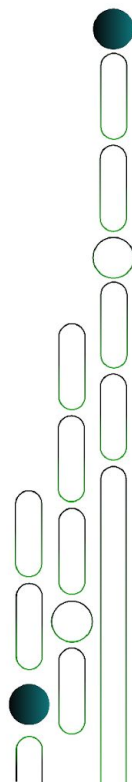
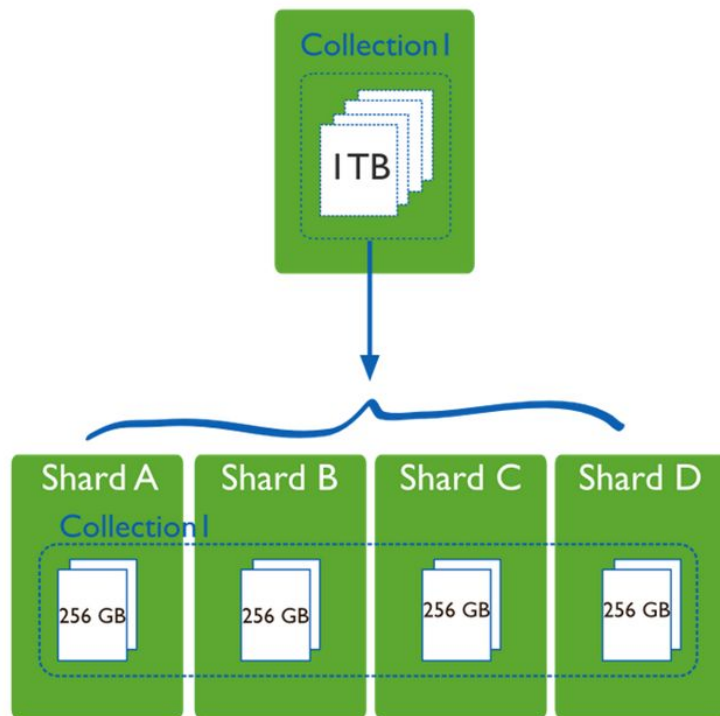
XPe



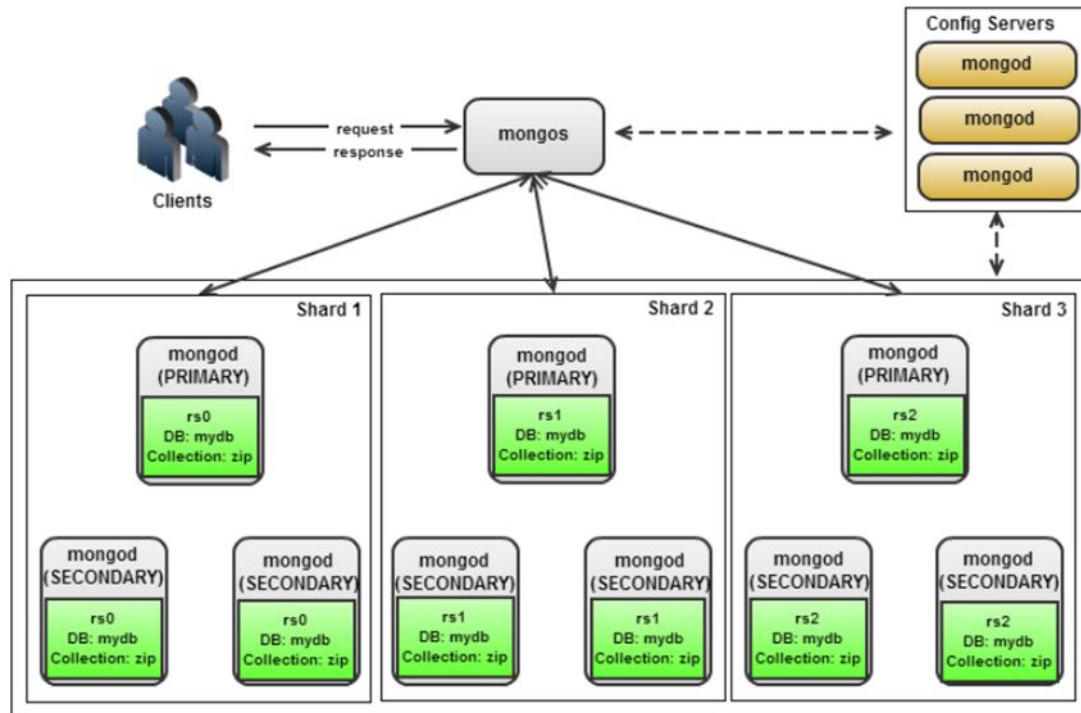
Replica Set



Sharding

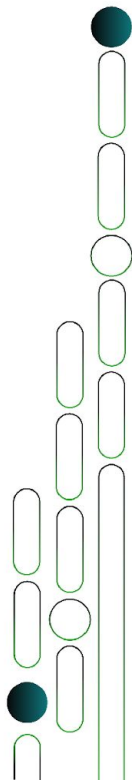


Arquitetura Mongodb



Data Driven

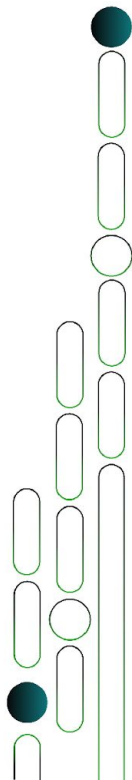
- O Data Driven se baseia no uso de ferramentas tecnológicas capazes de coletar e analisar dados diferentes da sua empresa.
- Esses dados, por sua vez, podem ser compilados por meio de BI ou inteligência artificial e ajudam o gestor a ter uma ideia mais precisa do seu negócio, facilitando a tomada de decisão estratégica.



Data Driven



- Assim, a gestão Data Driven é bem diferente dos modelos tradicionais, nos quais a tomada de decisão, geralmente, se baseava na intuição do dono ou nos “palpites” de especialistas, sem que houvessem dados reais para embasar essas atitudes.
- A maior dificuldade dos gestores em implementar a gestão Data Driven é, justamente, compreender a sua importância. Afinal, muitos acreditam que a tomada de decisão estratégica não faz tanta diferença nos resultados finais.



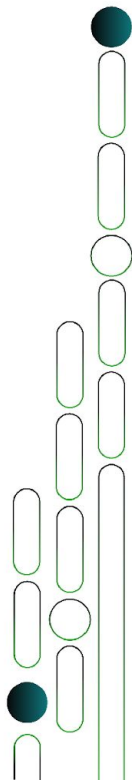
Como adotar o Data Driven?



O principal objetivo do Data Driven é entregar respostas mais precisas e confiáveis por meio de dados.

Por isso, o primeiro passo para implementar essa ideia é modificar a postura dos gestores e colaboradores, de forma que eles passem a reduzir os “achismos” e entendam o valor desses dados.

- Transforme a sua cultura.
- Use boas soluções.
- Aprenda a entender os dados.
- Use os indicadores de performance.

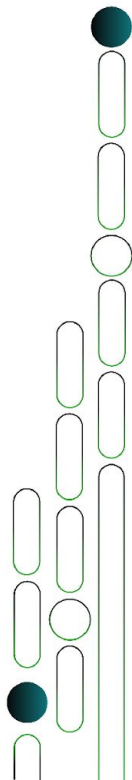


Data Driven



O Data Driven surgiu como extensão da Ciência de Dados, campo do conhecimento que utiliza métodos científicos e algoritmos para transformar dados – estruturados e não estruturados – em conhecimento.

Atualmente, no ambiente corporativo, isso é feito por meio de ferramentas como Big Data, Inteligência Artificial e Machine Learning, para obter insights a partir da coleta, cruzamento e interpretação de dados do mercado.

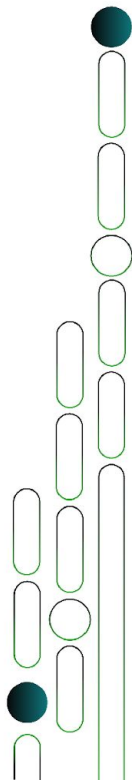


Data Driven x Analytics Driven

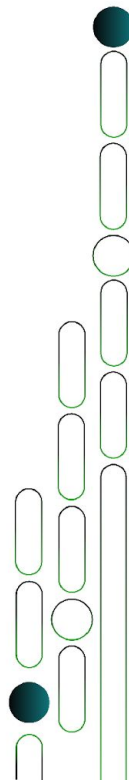
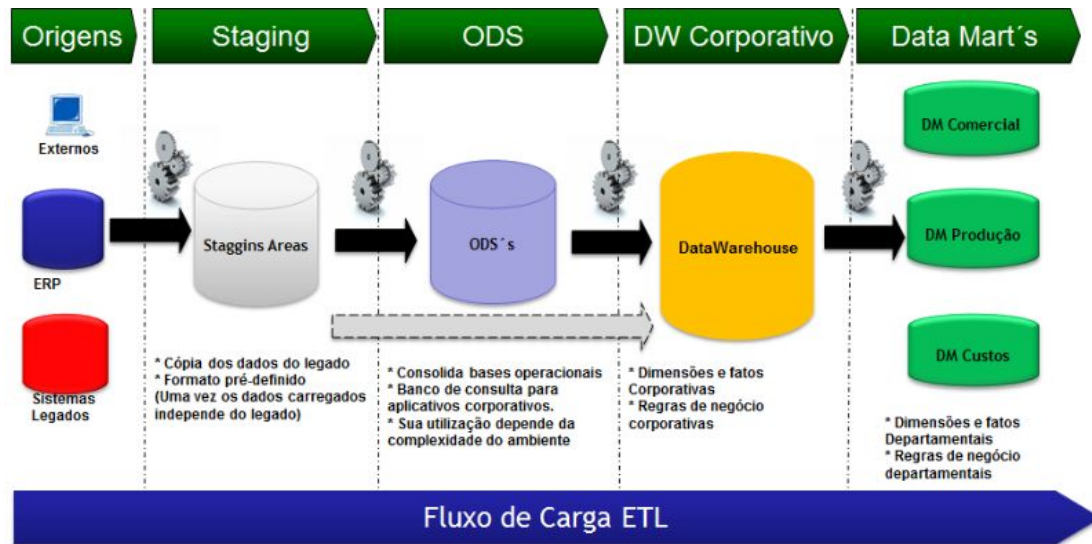


Embora ambos os conceitos derivem da Ciência de Dados, existem algumas diferenças fundamentais.

- O processo Data Driven possui uma abordagem mais quantitativa, uma vez que se baseia em números e modelos preditivos.
- O Analytics Driven também considera o aspecto qualitativo, estabelecendo padrões e correlações entre os dados. Podemos dizer que o Analytics Driven vai além da análise de dados, interpretando, também, o contexto e outras variáveis ligadas a essas informações que podem impactar nos resultados.



Business Intelligence

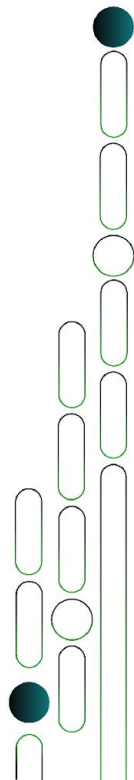


Por que NoSQL?



Hoje, as empresas estão adotando NoSQL para um número crescente de casos de uso. A escolha que é impulsionada por quatro megatendências inter-relacionadas:

- Big Users.
- Big Data.
- Internet das coisas.
- Cloud Computing.



Data Lake no noSQL



Um Data Lake pode residir em Hadoop, NoSQL, Amazon Simple Storage Service, Banco de Dados Relacional, ou combinações diferentes deles.

Alimentado por fluxos de dados (Data Streams).

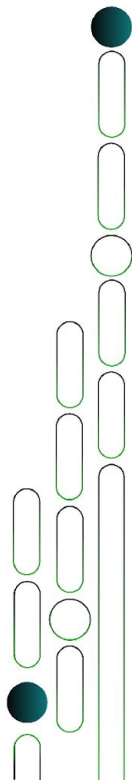
Data Lake tem muitos tipos de elementos de dados, estruturas de dados e metadados no HDFS, sem levar em conta a importância, IDs ou resumos e agregados.

Importante entender a natureza variada dos dados do Data Lake em relação ao banco de dados NoSQL:

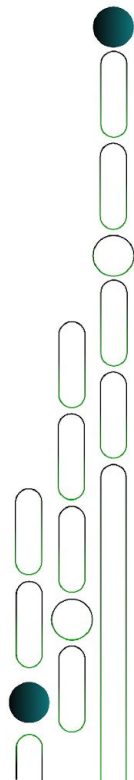
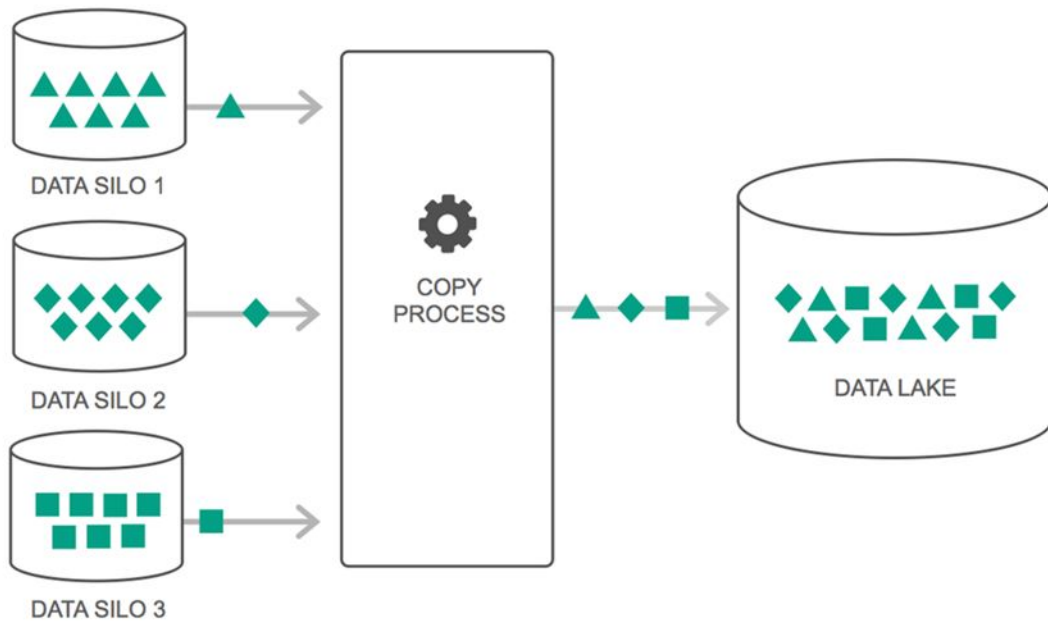
- Semiestruturado.
- Chave: valor (principalmente) com sua estrutura hierárquica.

Mais frequentemente, um Data Lake é mantido no Hadoop e alimentado de ou para NoSQL.

- NoSQL é um armazenamento de dados operacional, não analítico.



Data Lake



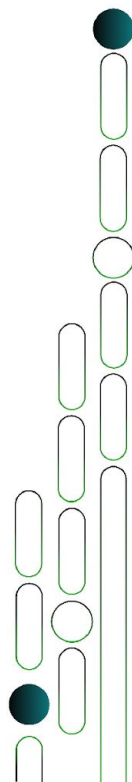
Big Data



Qual é o tamanho da web?



IoT - Internet of Things



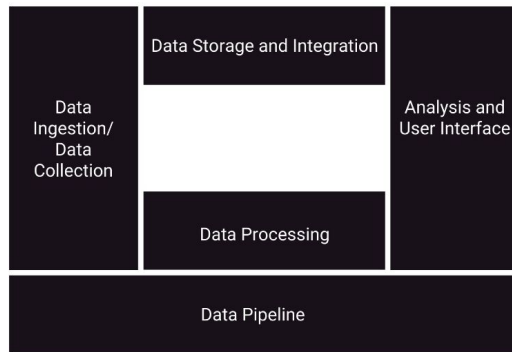
Camadas em uma Arquitetura de Dados



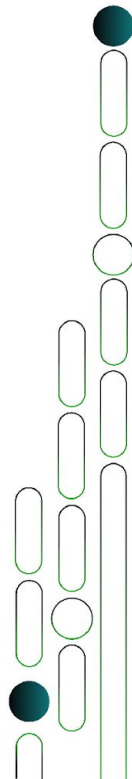
As diferentes camadas da arquitetura da plataforma de dados incluem:

- Camada de ingestão de dados.
- Camada de armazenamento de dados.
- Camada de processamento e análise de dados.
- Camada de interface do usuário.
- Camada de pipeline de dados.

Layers of the Data Platform Architecture



Fonte:
<https://www.analyticsvidhya.com>



Ingestão / Coleta de Dados

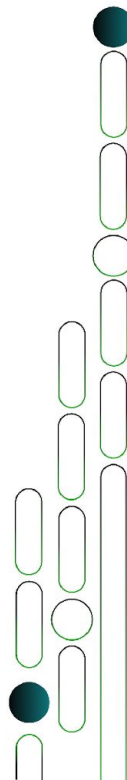
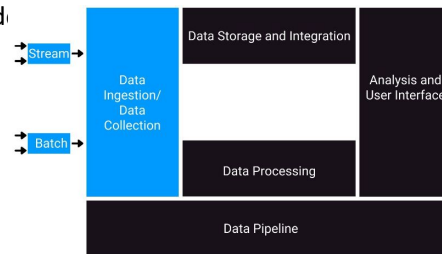


Esta camada executa as seguintes tarefas:

- É responsável pela conexão com as fontes de dados.
- É responsável pela transferência de dados das fontes de dados para a plataforma de dados no modo streaming, no modo batch ou em ambos.
- É responsável por manter as informações sobre os dados coletados no repositório de metadados.

Por exemplo, quantos dados são coletados na plataforma de dados e quantas informações descritivas?

Fonte: <https://www.analyticsvidhya.com>



Ingestão / Coleta de Dados



Existem várias ferramentas disponíveis no mercado.

Como ferramentas de podemos citar:

- Google Cloud Data Flow



- IBM Streams



- Amazon Kinesis

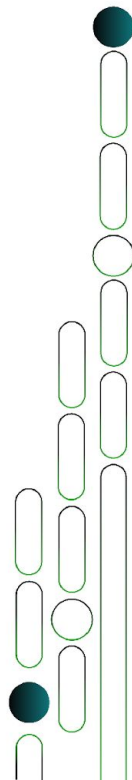


- Apache Kafka

São algumas das ferramentas mais usadas para ingestão de dados que suportam

modos batch e streaming.

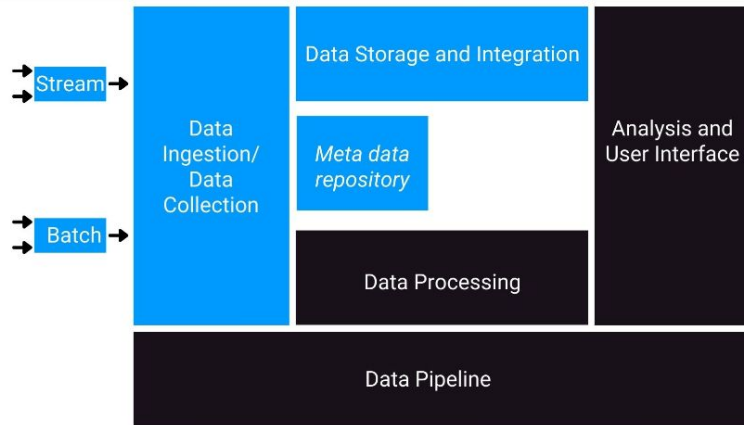
Fonte: <https://www.analyticsvidhya.com>



Armazenamento e Integração de Dados



As ferramentas para integração incluem o IBM Cloud Pak for Integration e o Open Studio da IBM. Uma vez que os dados tenham sido ingeridos, armazenados e integrados, eles precisam ser processados. Com isso, avançamos para a Camada de Processamento de Dados.



Fonte: <https://www.analyticsvidhya.com>

Armazenamento e Integração de Dados

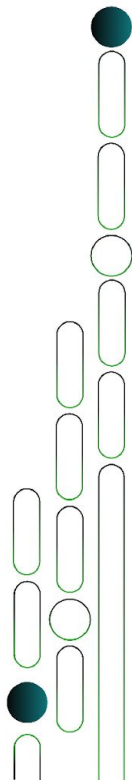


Por exemplo, o software de integração do Cloud Pak for Integration fornece um conjunto abrangente de ferramentas de integração, conectando aplicações e dados em qualquer ambiente de nuvem ou no local.

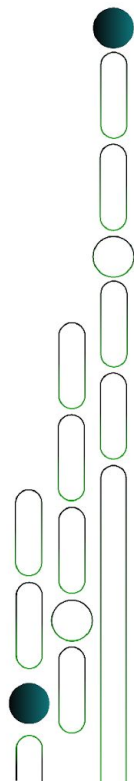
É uma plataforma cloud impulsionada por IA que permite a automação das operações de rede para que os provedores de serviços de comunicação (CSPs) possam fornecer serviços mais rapidamente.

A plataforma oferece interações de eventos em tempo real, transfere dados em qualquer nuvem, implementa e escala com arquitetura nativa em nuvem e disponibiliza serviços básicos compartilhados, tudo com segurança e criptografia de classificação corporativa de ponta a ponta.

O Cloud Pak for Integration usa serviços e recursos da AWS, inclusive Virtual Private Clouds (VPCs), zonas de disponibilidade, grupos de segurança, o Amazon Elastic Block Store (Amazon EBS), o Amazon Elastic Compute Cloud (Amazon EC2) e o Elastic Load Balancing para criar uma plataforma de nuvem confiável e



Data Driven – Importância da Comunicação



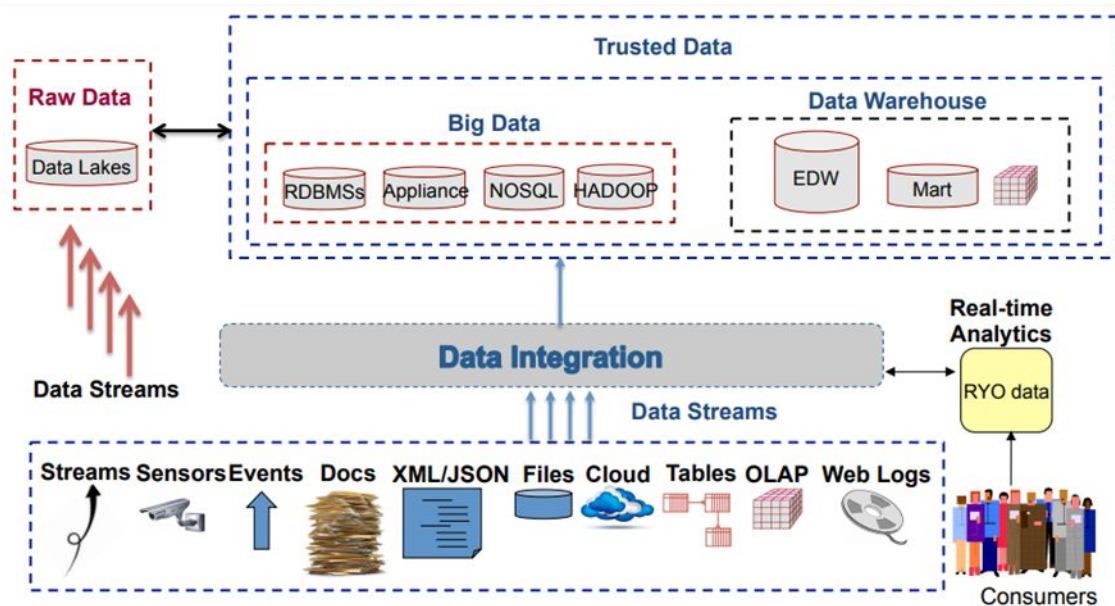
NoSQL e BI



Os bancos NoSQL ainda não oferecem ferramentas para BI.
Existem algumas abordagens para BI e NoSQL.



NoSQL e BI



Arquitetura HDFS



HDFS cluster possui 2 tipos de nodes:

Namenode (master node)
Datanode (worker node)

- Gerencia a estrutura do file system
- Gerencia os metadados de todos os arquivos e diretórios dentro da estrutura

Namenode

- Armazena e busca blocos de dados quando solicitado pelo cliente ou Namenode
- Reporta periodicamente para o Namenode com a lista de blocos que foram armazenados

Datanode

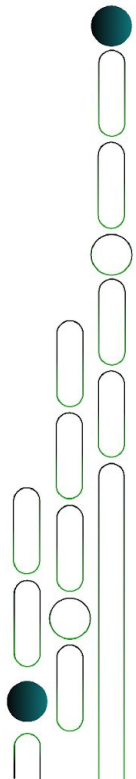


New SQL

ScaleArc AKTEAN ScaleDB SCHOOER

JustOneDB

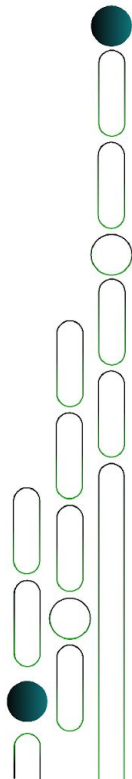
VoltDB xeround GENIEDB Clustrix SQLFire TRANSATTICE H-Store



NewSQL



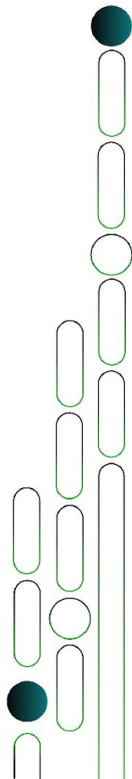
- Os bancos de dados NewSQL buscam promover a mesma melhoria de desempenho e escalabilidade dos sistemas NoSQL, não abrindo mão dos benefícios dos bancos de dados tradicionais, da linguagem SQL e das propriedades ACID.
- A vantagem dos bancos de dados NewSQL é proporcionar consultas em tempo real, além de maior capacidade de processamento. Há um custo grande nos bancos NoSQL em não usar SQL, sendo exigido trabalho excessivo dos desenvolvedores para compensar sua ausência.



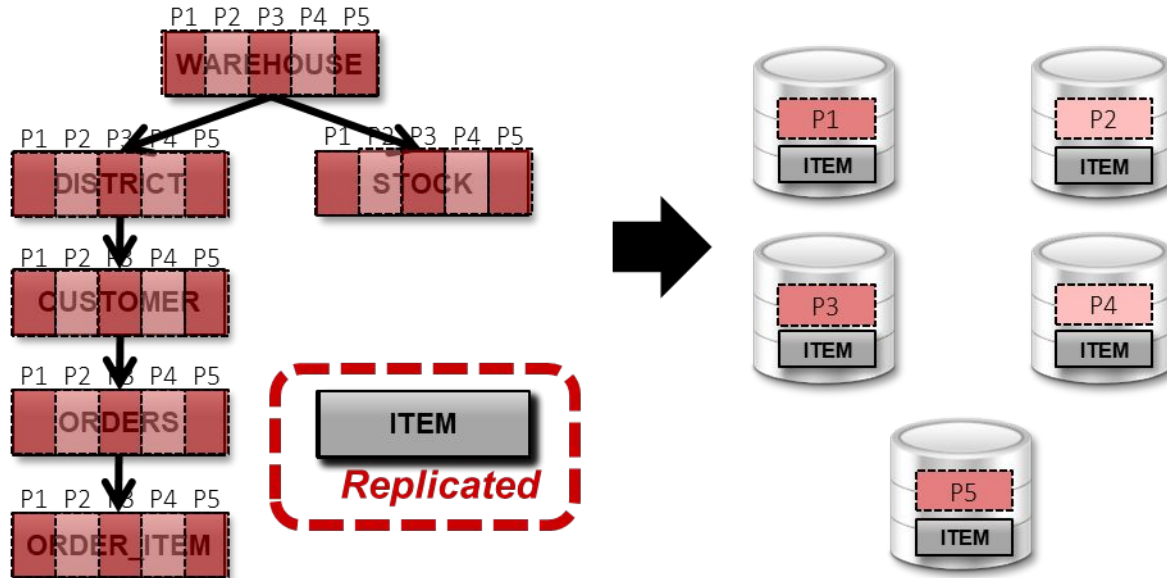
NewSQL



- Cinco características de um SGBD NewSQL:
- Linguagem SQL como meio de interação entre o SGBD e a aplicação.
- Suporte para transações ACID.
- Controle de concorrência não bloqueante, para que as leituras e escritas não causem conflitos entre si.
- Arquitetura que forneça um maior desempenho por nó de processamento.
- Arquitetura escalável, com memória distribuída e com capacidade de funcionar em um aglomerado com um grande número de nós.



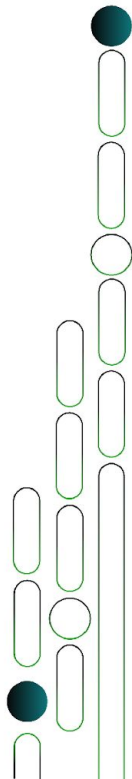
NewSQL - Database Partitioning



NewSQL



- MemSQL: Como o próprio nome sugere, é operado em memória e é um sistema de banco de dados de alta escala, por sua combinação de desempenho e compatibilidade com o SQL transacional e ACID na memória, adicionando uma interface relacional em uma camada de dados in-memory.
- VoltDB: Projetado por vários pesquisadores de sistema de banco de dados bem conhecidos, esse banco oferece a velocidade e a alta escalabilidade dos bancos de dados NoSQL, mas com garantias ACID, e sua latência em milissegundo e integração com Hadoop.
- SQLFire: Servidor de banco de dados NewSQL da VMware, desenvolvido para escalar em plataformas nas nuvens e tomar as vantagens de infraestrutura virtualizadas.
- MariaDB: Foi desenvolvido pelo criador do MySQL e é totalmente compatível com o MySQL. Também pode interagir com os bancos de dados NoSQL, como Cassandra e LevelDB.

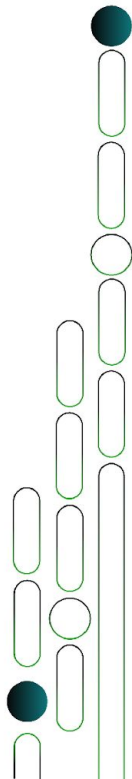


NewSQL, NoSQL e BigData

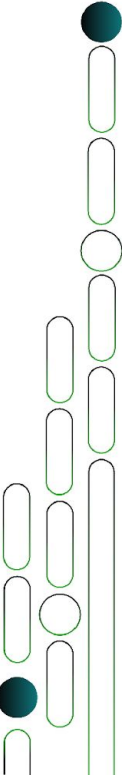
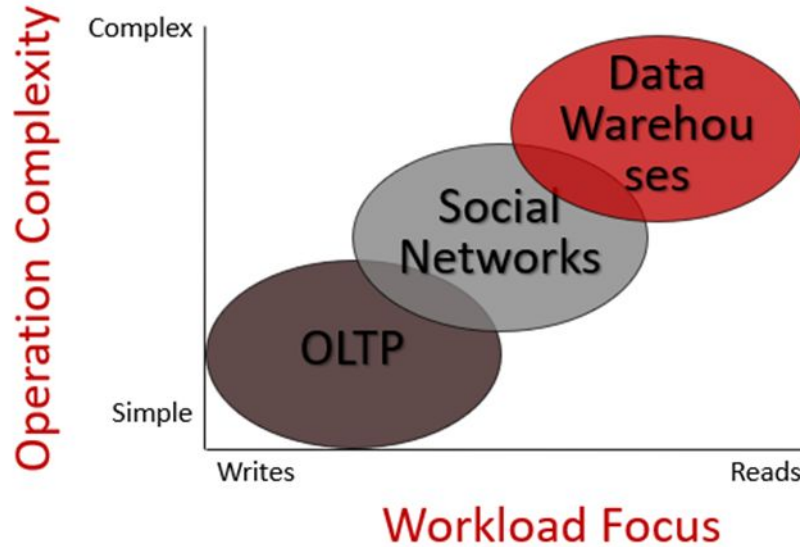


Algo interessante é que os bancos NoSQL e os NewSQL são concebidos e evoluídos como soluções para os problemas enfrentados pelo Big Data.

Como são poucas as empresas que sofrem com problemas de larga escala como as gigantes (Google, Facebook, Amazon etc.), o que impulsiona, de fato, as melhorias nos bancos de dados é o uso de soluções analíticas, como BI, Data Warehouse, ferramentas de ETL e de cubos.



NewSQL - Complexidade

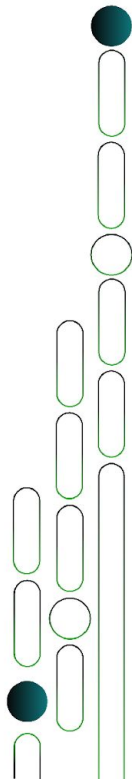


NewSQL



Podemos observar que o NewSQL é um aprimoramento do RDBMS utilizando o melhor dos dois mundos. Ele manteve totalmente as características dos bancos de dados padrões, implementando as inovações do NoSQL

Existem alguns bancos NoSQL especialistas com o poder do NewSQL, como é o caso do MariaDB (desenvolvido pelo criador do MySQL).



MariaDB - Tipos

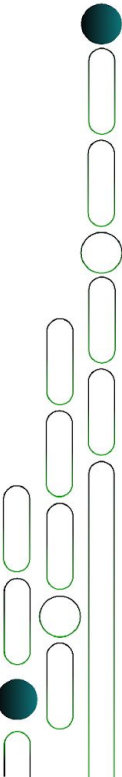


Date/Time Data Type	Description
Date	Displayed in the form 'yyyy-mm-dd.' Values range between '1000-01-01' and '9999-12-31'.
Datetime	Displayed in the form 'yyyy-mm-dd hh:mm:ss'. Values range between '1000-01-01 00:00:00' and '9999-12-31 23:59:59'.
timestamp(m)	Displayed in the form 'yyyy-mm-dd hh:mm:ss'. Values range between '1970-01-01 00:00:01' utc and '2038-01-19 03:14:07' utc.
Time	Displayed in the form 'hh:mm:ss'. Values range between '-838:59:59' and '838:59:59'.

Large Object Datatypes (LOB)

They include the following:

Large object Datatype	Description
tinyblob	Its maximum size is 255 bytes.
blob(size)	Takes 65,535 bytes as the maximum size.
mediumblob	Its maximum size is 16,777,215 bytes.
longtext	It takes 4GB as the maximum size.



Creating a Table

For you to be able to create a table, you must have selected a database. The table can be created using the CREATE TABLE statement. Here is the syntax for the command:

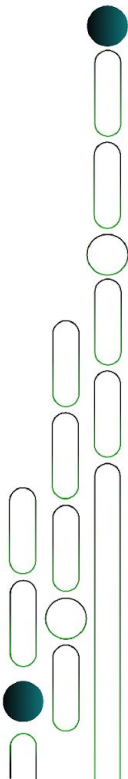
```
CREATE TABLE tableName (columnName columnType);
```

You can set one of the columns to be the primary key. This column should not allow null values.

We will create two tables within the Demo database, Book, and Price tables. Each table will have two columns.

Let's begin by creating the Book table with two columns, id and name. Run the following command:

```
CREATE TABLE Book(  
  id INT NOT NULL AUTO_INCREMENT,  
  name VARCHAR(100) NOT NULL,  
  PRIMARY KEY (id));
```



MariaDB



The SELECT statement helps us to view or see the contents of a database table. To see the contents of the Book table, for example, you need to run the following command:

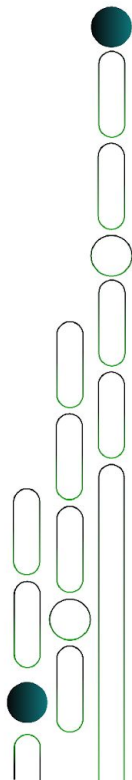
```
SELECT * from book;
```

```
MariaDB [Demo]> SELECT * from book;
+----+-----+
| id | name  |
+----+-----+
| 1  | MariaDB Book |
+----+-----+
1 row in set (0.001 sec)
MariaDB [Demo]>
```

Now, view the contents of the Price table:

```
SELECT * from price;
```

```
MariaDB [Demo]> SELECT * from price;
+----+-----+
| id | price |
+----+-----+
| 1  | 200   |
+----+-----+
1 row in set (0.001 sec)
MariaDB [Demo]>
```



INSERT

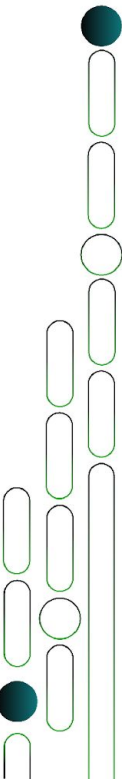
To insert data into a MariaDB table, you should use the INSERT INTO statement. This command takes the syntax given below:

```
INSERT INTO tableName
(column_1, column_2, ... )
VALUES
(value1, value2, ... ),
(value1, value2, ... ),
...;
```

The above syntax shows that you have to specify the table columns into which you want to insert data as well as the data that you need to insert.

Let us insert a record into the Book table:

```
INSERT INTO book
(id, name)
VALUES(1, 'MariaDB Book');
```



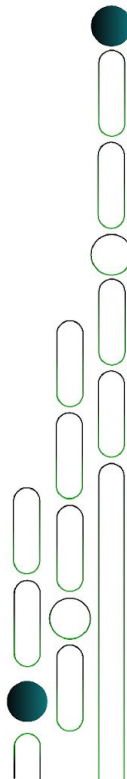
MariaDB



```
UPDATE book
SET name = "MariaDB Book1"
WHERE id = 1;
```

```
DELETE FROM tableName
[WHERE condition(s)]
[ORDER BY exp [ ASC | DESC ]]
[LIMIT numberOfRows];
```

```
DELIMITER $$
CREATE PROCEDURE myProcedure()
BEGIN
    SELECT name FROM book;
END;
;
```

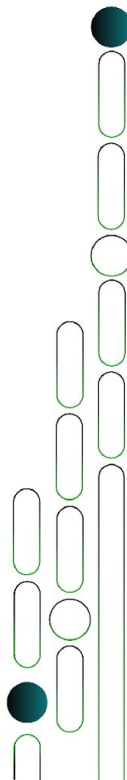


Cockroach DB



CockroachDB é um banco de dados de código aberto SQL distribuído e construído em um armazenamento de chave-valor transacional e fortemente consistente desenvolvido pela Cockroach Labs.

- Dimensionado horizontalmente;
- Sobrevive a falhas de disco, máquina, rack e até mesmo de datacenter com o mínimo de interrupção e intervenção manual;
- Suporta transações ACID fortemente consistentes;
- Fornece uma API SQL familiar para estruturar, manipular e consultar dados.

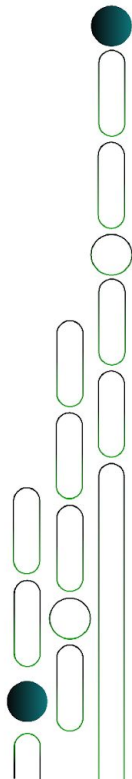


Cockroach DB



As principais características do CockroachDB são:

- Compatível com SQL - Embora o CockroachDB tenha um armazenamento de chave-valor transacional distribuído e fortemente consistente por baixo, sua API externa é compatível com o SQL padrão.

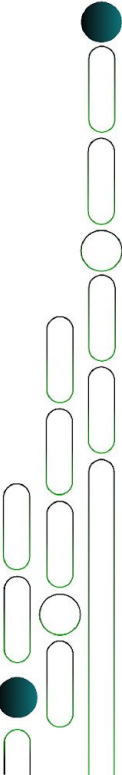


Cockroach DB



As principais características do CockroachDB são:

- Disponibilidade múltipla ativa - O modelo de disponibilidade do CockroachDB é denominado como "disponibilidade multi-ativa". A disponibilidade multi-ativa oferece benefícios de leitura e gravação de cada nó em um cluster sem conflitos. Várias réplicas executam serviços idênticos, e o tráfego é roteado para todos eles. Se alguma réplica falhar, as outras simplesmente lidam com o tráfego que seria roteado para ela.



Cockroach DB



As principais características do CockroachDB são:

- Alterações de esquema on-line - CockroachDB fornece um recurso integrado de alterações de esquema on-line; uma maneira simples de atualizar um esquema de tabela sem impor quaisquer consequências negativas em um aplicativo. As alterações no esquema da tabela acontecem enquanto o banco de dados está em execução. A alteração do esquema é executada como um processo em segundo plano sem travar os dados da tabela subjacente. Isso permite que as consultas do aplicativo sejam executadas normalmente sem qualquer efeito na leitura/gravação.

