

# Banco de Dados

Profa. Dra. Vanessa Borges



# Módulo 4 - SGBDs NoSQL



# Módulo IV – SGBDs NoSQL

- Unidade 1 - Introdução ao NoSQL
- Unidade 2 - NoSQL orientado a documentos
- Unidade 3 - Outros SGBDs
- Objetivo
  - Conhecer princípios e a aplicação prática de sistemas gerenciadores de banco de dados NoSQL

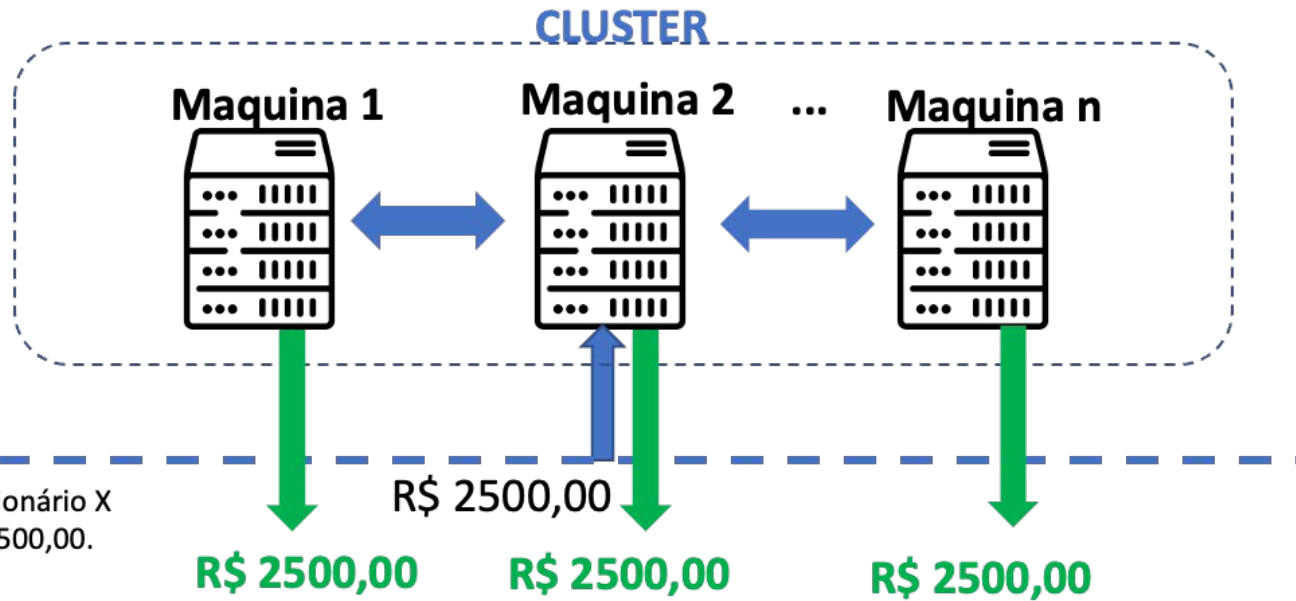
# Banco de Dados NoSQL

- NoSQL (*Not Only SQL* – não somente SQL) é um termo genérico definido para uma classe de banco de dados não relacionais (ACID)
- O termo iniciou em 1998 como o nome de um banco de dados relacional de código aberto que não possuía uma interface SQL
- Em 2009 o termo foi utilizado por Eric Evans como nome de um evento para discutir bancos de dados open source distribuídos
- Essa foi uma tentativa de descrever o surgimento de um número crescente de banco de dados não relacional que não tinham a preocupação de fornecer garantias ACID

# Relacional vs NoSQL

	Banco de dados relacional	Banco de dados NoSQL
Característica	Tabelas, esquema definido, redundância mínima, entidade e relacionamento, ACID	Registros, não possui estrutura fixa, escalabilidade
Necessidades	Sistemas financeiros, sistemas corporativos, consistência dos dados	Sistemas em nuvem, análises sociais, performance na consulta/escrita, replicação
SGBD	DB2, Firebird, InterBase, SQL Server, MySQL, Oracle, PostgreSQL	Cassandra, BigTable, MongoDB, CouchDB, Dynamo
Empresas que utilizam	SAP, OpenERP, Previdência Social, Caixa, Itaú	Twitter, Facebook, Amazon, LinkedIn, Google, Yahoo
Consistência	As regras de integridade garantem a consistência das informações. <b>Consistência rígida</b>	Não garante a consistência da informação, caso nenhuma informação seja atualizada, retornará o mesmo valor para todas as solicitações. <b>Consistência eventual</b>

# Consistência Rígida



Se houver uma alteração de salário na **Máquina 2**, essa alteração só será persistida **quando todos os nós do cluster forem atualizados**

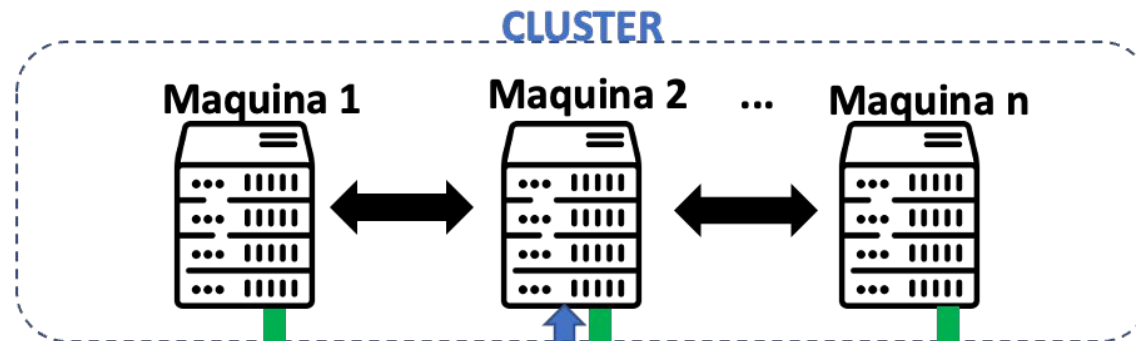
- **Demanda grande esforço de replicação de dados entre os nós do cluster**
  - **Problemas de escalabilidade**

- **Visão única dos dados**
- **Dados CONSISTENTES**



**Banco de dados relacional**

# Consistência Eventual (assíncrono)



Se houver uma alteração de salário na **Máquina 2**, essa alteração **pode** ser dita **concluída**

- **Replicação dos dados entre nós do cluster de forma assíncrona**
  - **Simplifica escalabilidade**

**Tempo 1:**  
Altere o salário do funcionário X de R\$2000,00 para R\$2500,00.

R\$ 2500,00

Milissegundos

**Tempo 2:**

R\$ 2000,00

R\$ 2500,00

R\$ 2000,00

**Tempo 3:**

R\$ 2500,00

R\$ 2500,00


R\$ 2500,00



Banco de dados nosql

# Categorias de NoSQL

**Key Value**



**Example:**  
Riak, Tokyo Cabinet, Redis  
server, Memcached,  
Scalaris

**Document-Based**



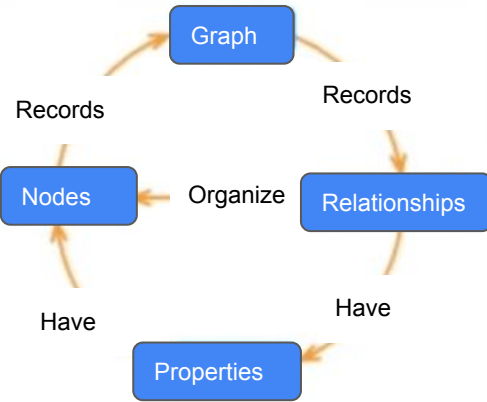
**Example:**  
MongoDB, CouchDB,  
OrientDB, RavenDB

**Column-Based**



**Example:**  
BigTable, Cassandra,  
Hbase,  
Hypertable

**Graph-Based**




**Example:**  
Neo4J, InfoGrid, Infinite  
Graph, Flock DB



# MongoDB - orientada a documentos

Key Value



Example:  
Riak, Tokyo Cabinet, Redis  
server, Memcached,  
Scalaris

Document-Based




Example:  
MongoDB, CouchDB,  
OrientDB, RavenDB

Column-Based



Example:  
BigTable, Cassandra,  
Hbase,  
Hypertable

Graph-Based



Example:  
Neo4J, InfoGrid, Infinite  
Graph, Flock DB

# Principais diferenças entre relacional e orientado a documentos

Banco de dados Relacional		MongoDB
Armazenam dados em tabelas com linhas divididas em colunas	➡	Armazenam os dados em coleções de documentos com campos e subdocumento
Possuem relacionamento entre diferentes tabelas e registros	➡	Não possuem relacionamentos entre coleções e/ou documentos
Foco em garantir o ACID	➡	Foco em garantir escalabilidade, alta-disponibilidade e performance
Uso de transações para garantir <i>commits</i> e <i>rollbacks</i>	➡	Não há uso de transações e o ACID é garantido apenas a nível de documento. Transações devem ser garantidas a nível de aplicação
Uso da linguagem de consulta SQL para manipular o banco	➡	Uso de comandos JavaScript para manipular o banco
Cada coluna de uma linha pode armazenar apenas um dado (atômico)	➡	Cada campo de um documento pode armazenar múltiplos valores ou até mesmo outros documentos
Possuem chave-estrangeira e JOINS	➡	Não possuem chave-estrangeira nem JOINS
Schema pré-definido e rígido	➡	Schema variável conforme o uso da aplicação
Foco em não repetição de dados	➡	Foco em acesso rápido de dados
Menor consumo de disco	➡	Maior consumo de disco (geralmente)
Menor consumo de memória	➡	Maior consumo de memória (geralmente)

# Escolha do modelo relacional versus nosql

- Quando utilizar NoSQL?
- Indicado para aplicações que:
  - Irão trabalhar com grande volume de dados
  - Necessitam de velocidade de resposta (leitura e escrita) – escala horizontal
  - Não necessita de consistência rígida

# Módulo IV – SGBDs NoSQL

- Unidade 1 - Introdução ao NoSQL
- Unidade 2 - NoSQL orientado a documentos
- Unidade 3 - Outros SGBDs
- Objetivo
  - Conhecer princípios e a aplicação prática de sistemas gerenciadores de banco de dados NoSQL

# Licenciamento



Respeitadas as formas de citação formal de autores de acordo com as normas da ABNT NBR 6023 (2018), a não ser que esteja indicado de outra forma, todo material desta apresentação está licenciado sob uma [Licença Creative Commons - Atribuição 4.0 Internacional](https://creativecommons.org/licenses/by/4.0/).