

# Sistemas NoSQL - Colunas



# 2.5

quintilhões de bytes de dados gerados por dia na Web

# Motivação

- **Crescimento exponencial de dados na Web**
  - Dados semi ou não-estruturados
- **Sistemas de banco de dados relacionais pouco flexíveis**
  - Dados centralizados
- **Requisitos diferenciais das aplicações**
  - Escalabilidade sob demanda
  - Alto grau de disponibilidade



# O que é NoSQL?

**Termo genérico para designar qualquer banco de dados não-relacional**

# NoSQL = Not Only SQL

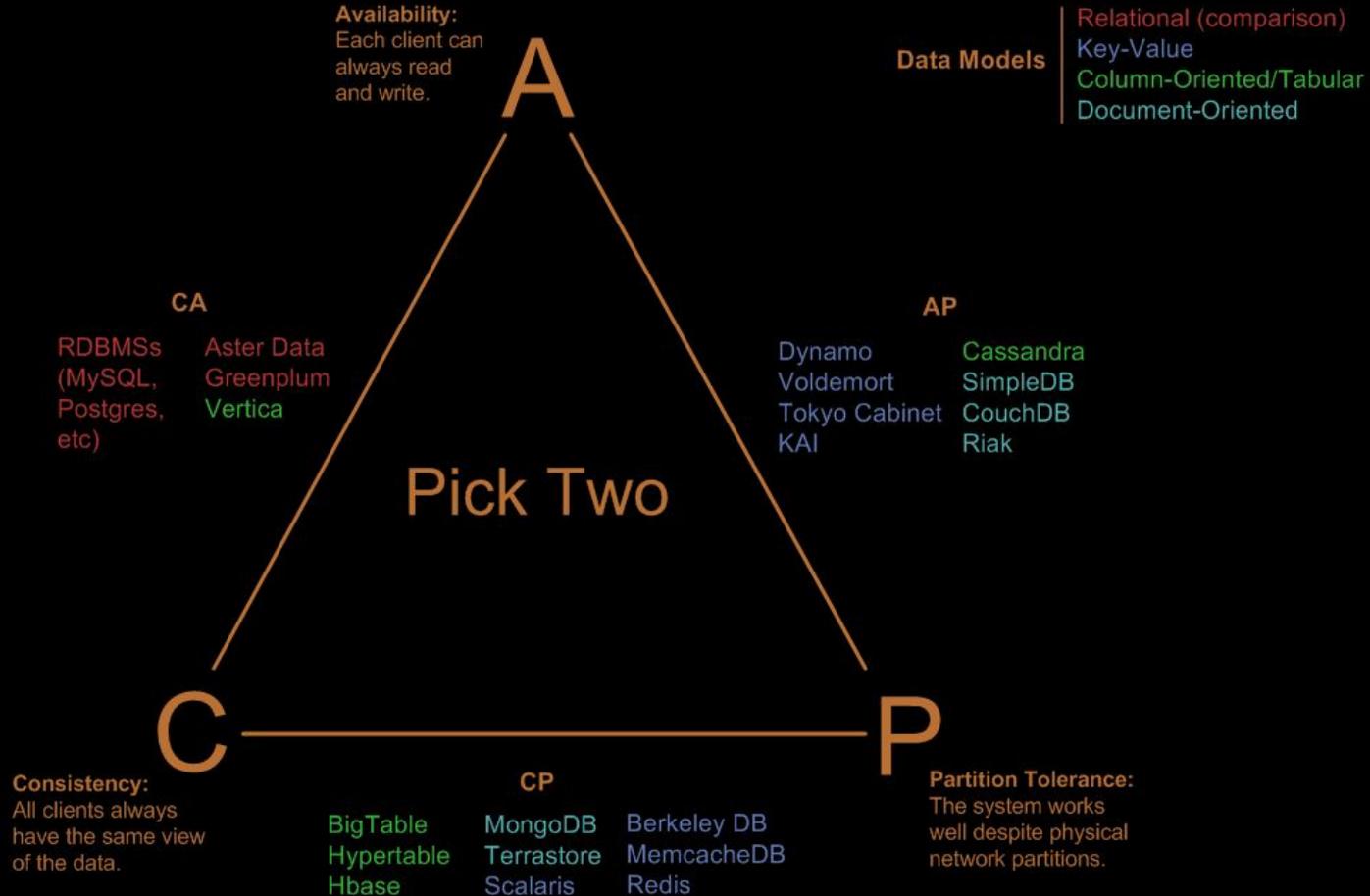
- Criados para gerenciar grande volume de dados com alto desempenho
- Resolver o problema de escalabilidade dos bancos tradicionais



# Teorema CAP

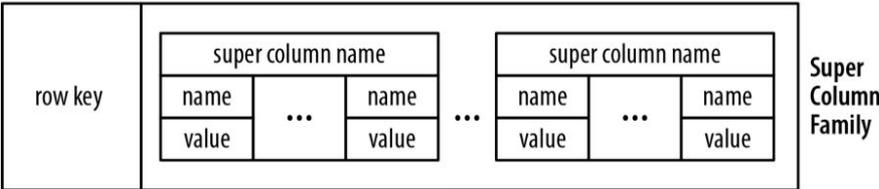
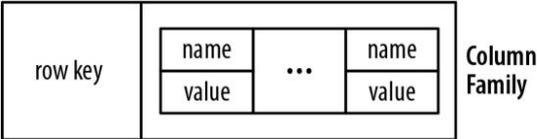
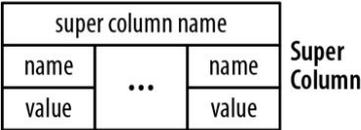
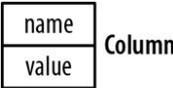
- **Sistemas de banco de dados distribuídos só garantem duas das três propriedades**
    - **Consistência:** todos os clientes tem a mesma visão dos dados
    - **Disponibilidade:** cada cliente tem direito à escrita e à leitura dos dados
    - **Tolerância à partição:** o sistema total mantém suas propriedades mesmo quando ocorre deploy em servidores diferentes, sendo transparente a todos os clientes
- 

# Visual Guide to NoSQL Systems



# NoSQL Orientados à Colunas

- Modelo baseado em atributos (colunas)
- Índice do dado composto por: linha, coluna e timestamp
- Operações de leitura e escrita são atômicas



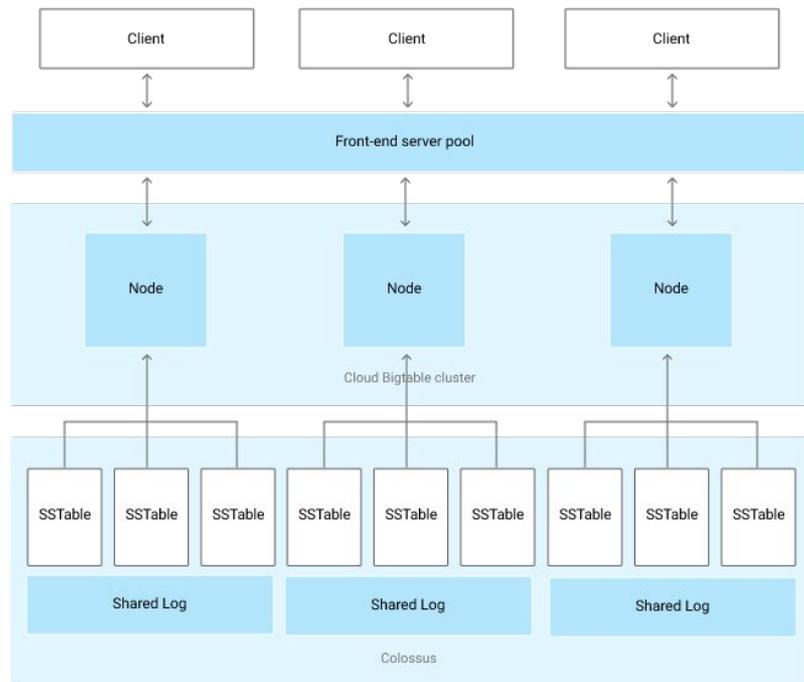
# Big Table



- **Surgiu em 2004, pela Google**
  - Projetada para armazenar dados em grande escala (petabyte de dados)
  - Permite particionamento de dados e forte consistência
  - Organizado em linha, coluna e timestamp
    - Linha - tablets
    - Colunas - famílias
    - Timestamp - versionamento de célula

# Big Table

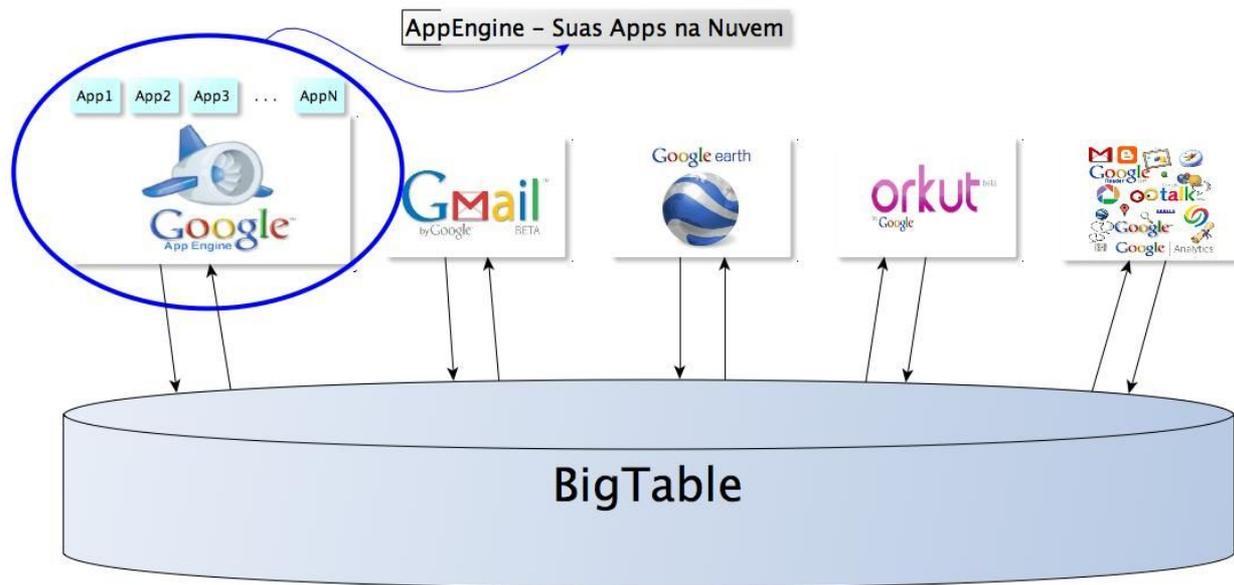
- **Arquitetura**



# Big Table

- Usado em mais de 60 produtos

Google™  
BigTable

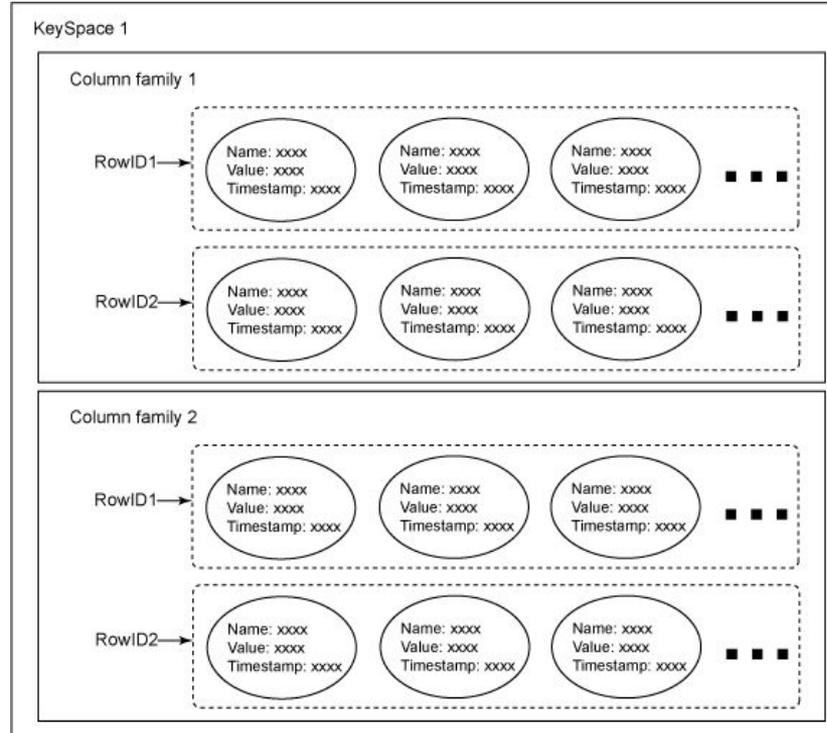


# Cassandra



- **Surgiu em 2008, pelo Facebook**
  - Descontinuado em 2010
- **Dynamo + BigTable**
- **Alta escalabilidade e disponibilidade, sem um ponto único de falha**
- **Organizados em linhas, colunas, famílias de colunas e keyspaces**

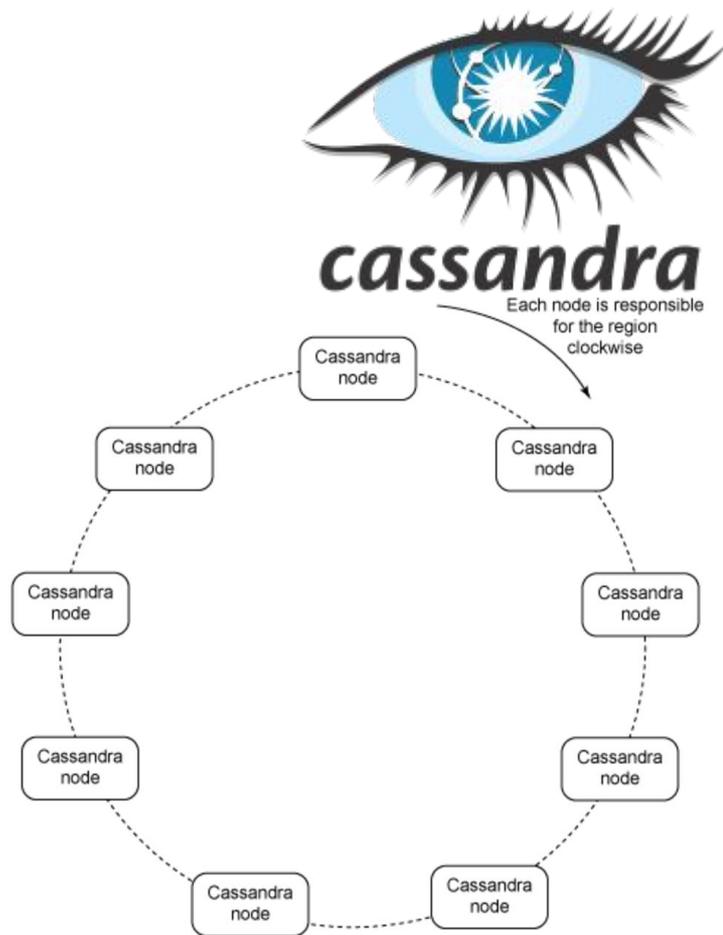
# Cassandra



# Cassandra

- **Arquitetura**

- Distribui os dados entre os nós
- Qualquer nó pode aceitar qualquer solicitação (leitura ou escrita)
- Escolha do nível de consistência
- Fácil para adicionar ou remover nós



# Cassandra

- Quem usa?



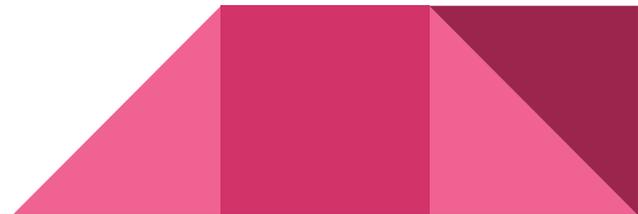
# Ranking - BD

8 systems in ranking, November 2016

Rank			DBMS	Database Model	Score		
Nov 2016	Oct 2016	Nov 2015			Nov 2016	Oct 2016	Nov 2015
1.	1.	1.	Cassandra 	Wide column store	133.97	-1.09	+1.05
2.	2.	2.	HBase	Wide column store	58.74	+0.54	+2.28
3.	3.	3.	Accumulo	Wide column store	3.47	+0.04	-0.47
4.	4.	4.	Hypertable	Wide column store	0.52	-0.03	-0.18
5.	5.		Google Cloud Bigtable	Wide column store	0.31	-0.04	
6.	6.	 5.	Sqrrl 	Multi-model	0.28	+0.02	-0.14
7.	7.		ScyllaDB	Wide column store	0.21	+0.06	
8.			MapR-DB	Multi-model 	0.15		

# Conclusão

- **Não substitui, mas complementa os bancos de dados relacionais**
- **Agrupamento de colunas otimiza a recuperação e armazenamento dos dados**
- **São mais adequados para sistemas OLAP**



The background is a solid pink color. In the top right corner, there are several overlapping geometric shapes in various shades of pink and magenta, including triangles and squares, creating a modern, abstract design.

Dúvidas?