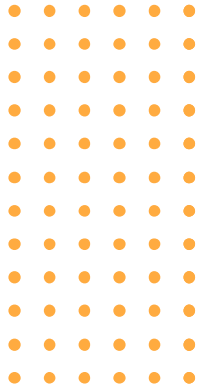


# Cassandra

Nomes: Daniela, Douglas e Flavia





1. O que é o Cassandra
2. Arquitetura
3. Curiosidades e quando usar
4. Instalação
5. Aplicação





01.

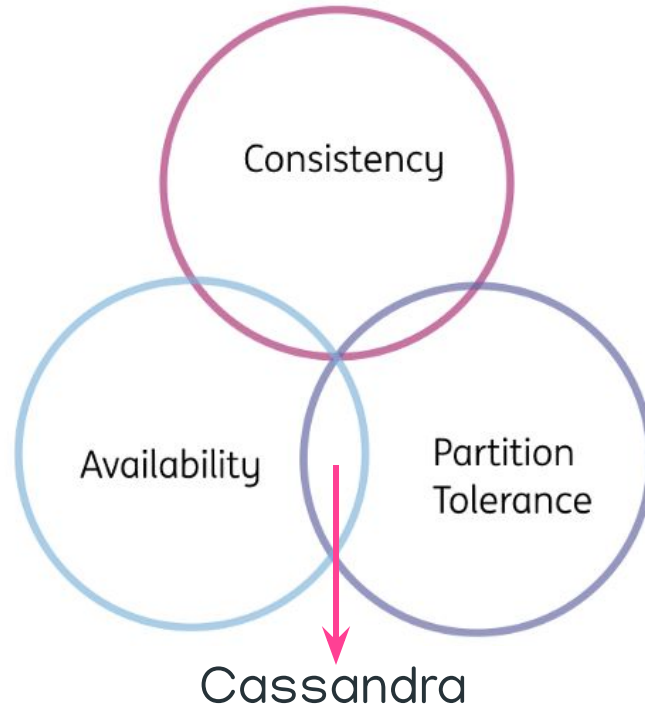
Mas quem é essa  
tal de  
Cassandra?

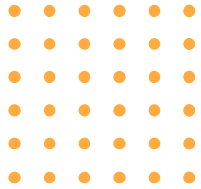
# O Cassandra é um banco NoSQL, open source:

- Rápido
- Distribuído
- Colunar
- Suporta MapReduce
- Suporta grande volume de dados
- Tolerante a falhas
- Altamente escalável



# Teorema CAP





# CQL

O dados são manipulados através da linguagem CQL (Cassandra Query Language) que pode ser executada no CQLSH

- `SELECT * FROM brinks.brinks_stock WHERE title = 'Brinquedo X'`  
`ALLOW FILTERING`
- `UPDATE brinks.brinks_stock SET stock = 5 + stock WHERE id = 12`

# Criação

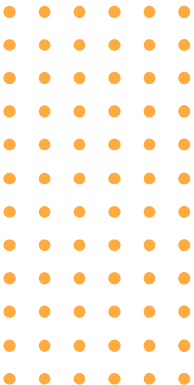



# Como assim colunar?


Orientado a Linhas	Orientado a Colunas
Joao   2432.00   1988   Rio de Janeiro	Joao   Maria   Pedro   Jorge
Maria   2511.00   1986   São Paulo	2432.00   2511.00   3500.00   4200.00
Pedro   3500.00   1976   Mato Grosso	1988   1986   1976   1930
Jorge   4200.00   1930   Paraná	Rio de Janeiro   São Paulo   Mato Grosso   Paraná

(a) (b)



Id	Name	Age	Gender	Car
1	{Name : Brian}	{Age : 21}	{Gender : M}	{Car : BMW}
2	{Name : John}	{Age : 43}	{Gender : M}	{Car : BMW}
3	{Name : Bob}	{Age : 45}	{Gender : M}	{Car : BMW}
4	{Name : Frank}	{Age : 23}	{Gender : M}	{Car : Audi}
5	{Name : Olivia}	{Age : 35}	{Gender : F}	{Car : Audi}
6	{Name : Emma}	{Age : 32}	{Gender : F}	{Car : Audi}
7	{Name : Sophia}	{Age : 45}	{Gender : F}	
8	{Name : Mia}	{Age : 23}	{Gender : F}	

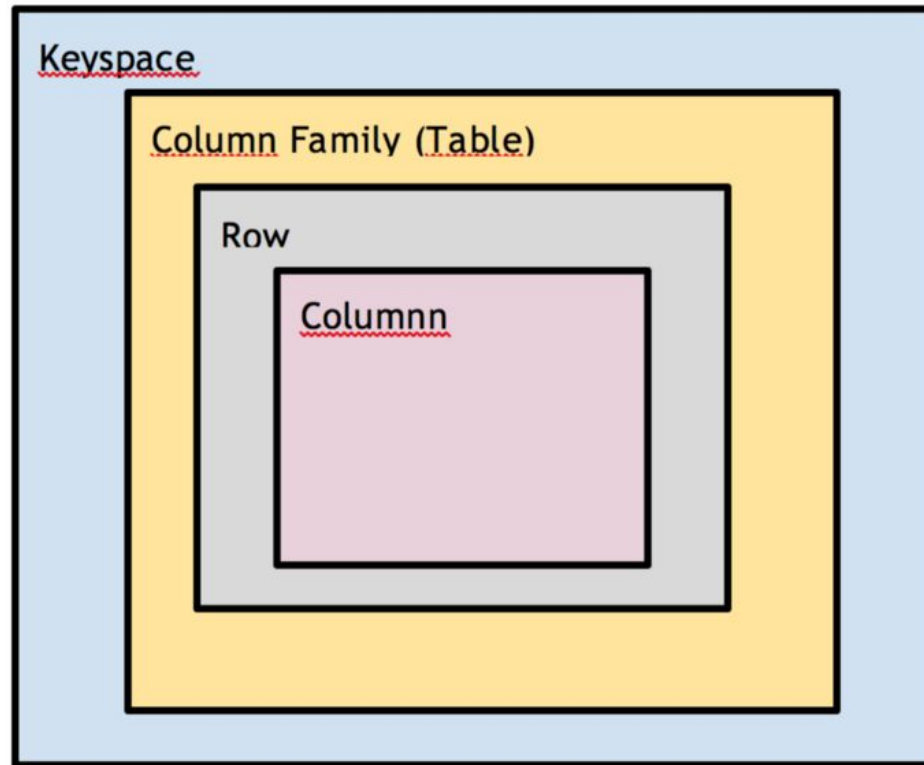


Id	Name	Age	Gender	Car
1	{Name : Brian}	{Age : 21}	{Gender : M} * 4	{Car : BMW}
2	{Name : John}	{Age : 43}		{Car : BMW}
3	{Name : Bob}	{Age : 45}		{Car : BMW}
4	{Name : Frank}	{Age : 23}		{Car : Audi}
5	{Name : Olivia}	{Age : 35}	{Gender : F} * 4	{Car : Audi}
6	{Name : Emma}	{Age : 32}		{Car : Audi}
7	{Name : Sophia}	{Age : 45}		
8	{Name : Mia}	{Age : 23}		



# 02.




## Arquitetura do Cassandra

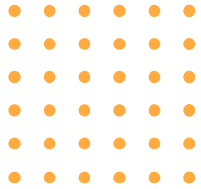




# Keyspace



- Podem ser comparado com Schemas quando comparados a um banco relacional
  - Contém conjuntos de Tabelas
- 
- 
- 



# Família de colunas (Tabela)

- É equivalente à uma tabela em bancos relacionais
- Um **Keyspace** → pode conter “N” Tabelas  
Uma **Tabela** → pode conter “N” Linhas

# Linhas

Linhas = Primary key + conjunto de colunas

O Cassandra mantêm apenas os campos que possuem dados, assim Linhas podem ter o número de colunas diferentes

**ROW 1**


**ROW 2**


**ROW 3**


# Colunas

Composta por:

- **Column key:** nome da coluna
- **Column value:** valor que está sendo persistido
- **Timestamp:** O Cassandra utiliza esse campo para resolver conflitos e determinar qual é o valor mais atual

# Transações

- **Atômico:** Tudo em uma transação é bem-sucedido ou toda a transação é revertida.
- **Isolado:** As transações isoladas (delete) não são visíveis para outros clientes até que termine.
- **Durável:** As transações concluídas persistem no caso de travamentos ou falha do servidor.
- Quando uma solicitação é enviada para qualquer nó, esse nó é eleito como nó coordenador, sendo responsável por gerenciar todo o caminho da solicitação e responder de volta para o cliente.

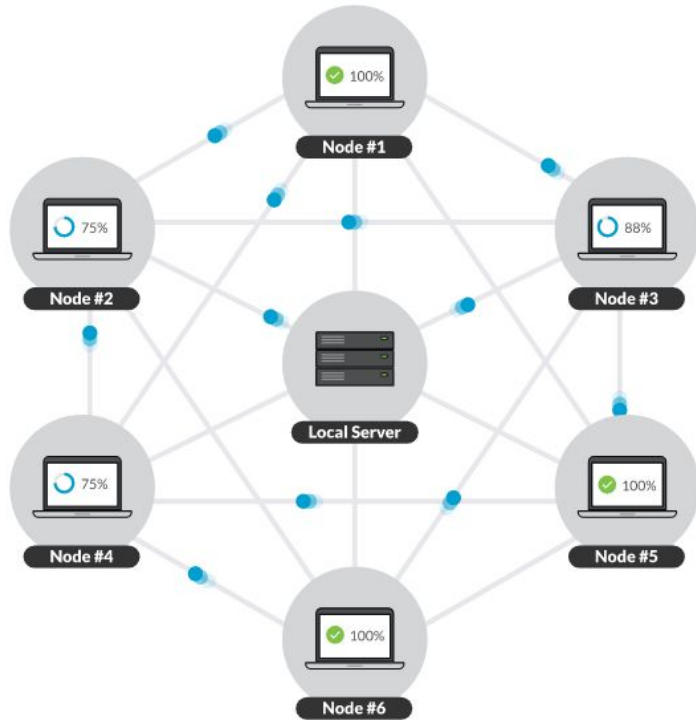


# Gossip (fofoca)

O Cassandra utiliza um protocolo chamado *Gossip* (fofoca) com o intuito de detectar problemas com os nós que funciona da seguinte forma:

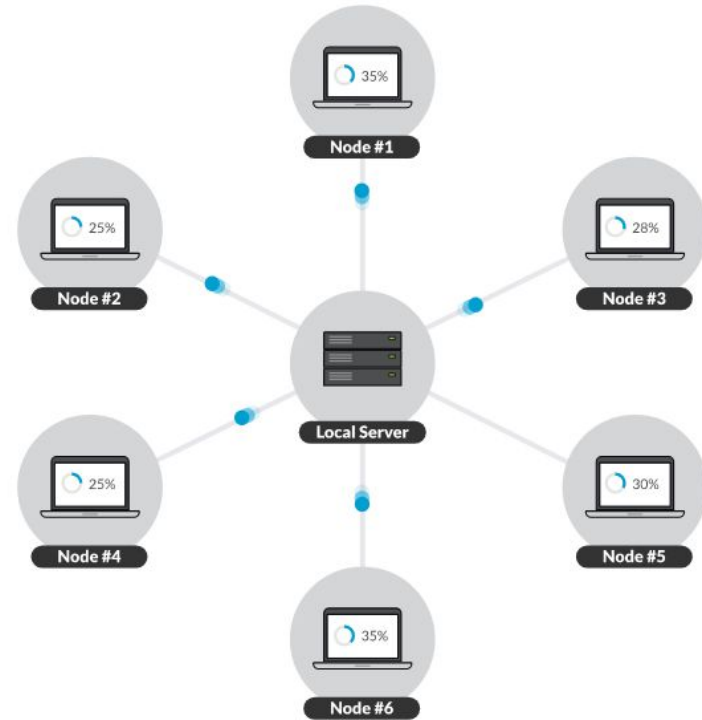
- Periodicamente os nós do cluster enviam mensagens entre eles sobre o seu estado e o estado dos outros nós que eles já sabem
- Caso algum nó deixe de responder ele é marcado e ações corretivas são acionadas

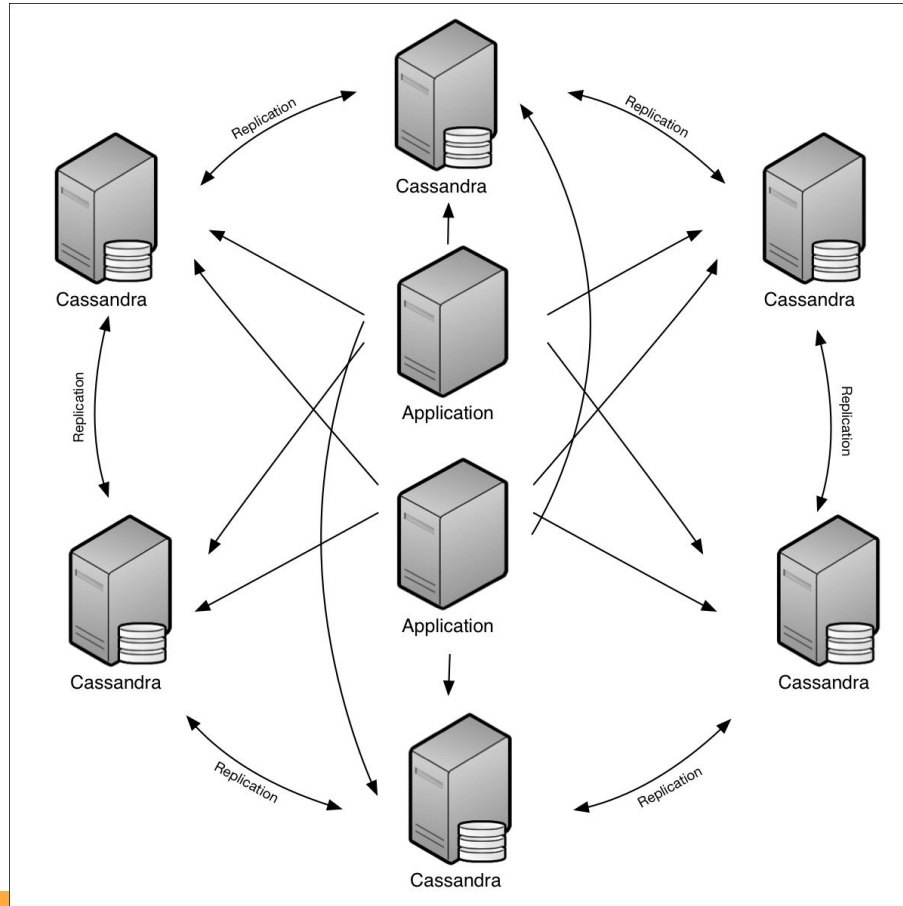
## Peer-to-Peer Architecture



Vs.

## Client-Server Architecture







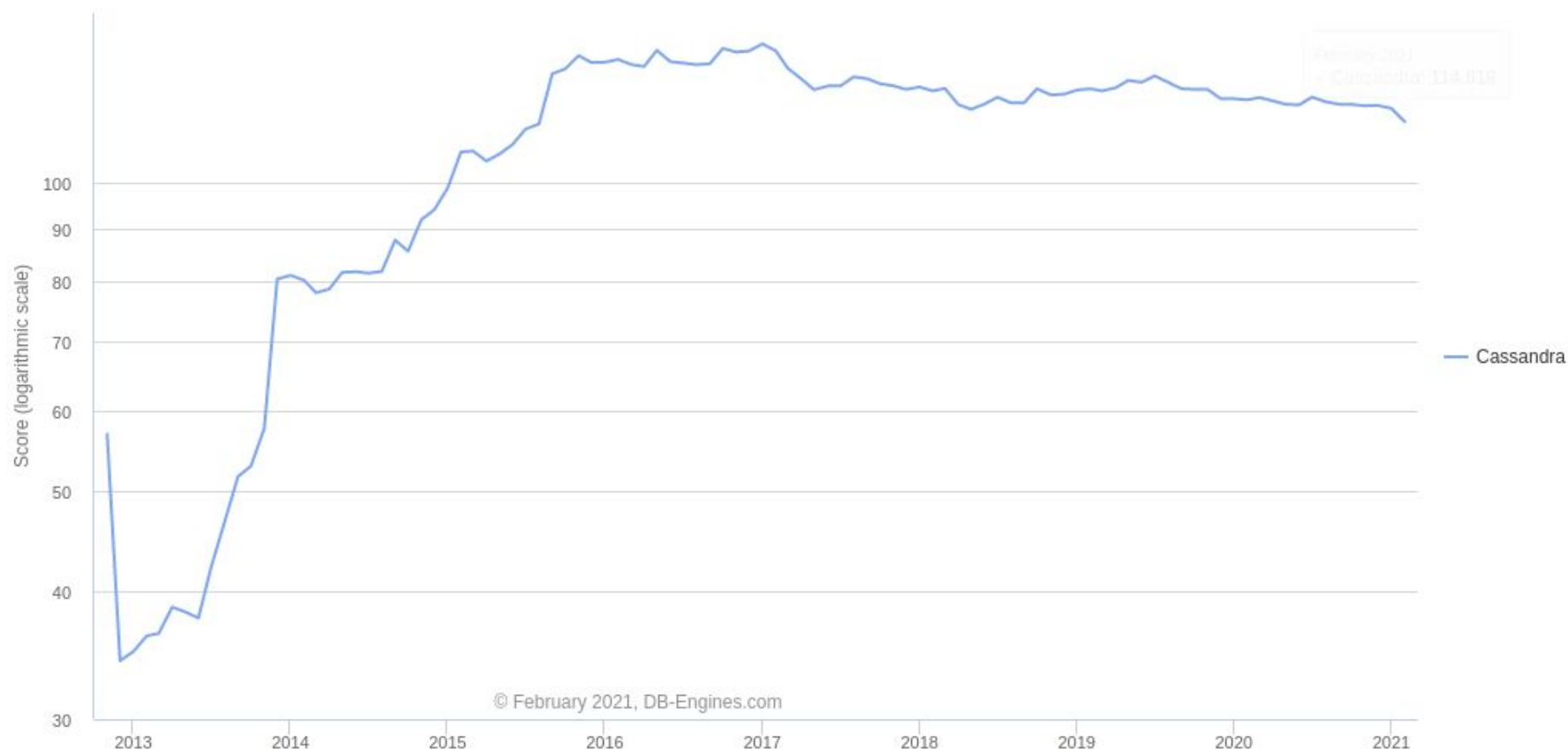
03.

# Curiosidades



Rank			DBMS	Database Model	Score		
Feb 2021	Jan 2021	Feb 2020			Feb 2021	Jan 2021	Feb 2020
1.	1.	1.	Oracle	Relational, Multi-model	1316.67	-6.26	-28.08
2.	2.	2.	MySQL	Relational, Multi-model	1243.37	-8.69	-24.28
3.	3.	3.	Microsoft SQL Server	Relational, Multi-model	1022.93	-8.30	-70.81
4.	4.	4.	PostgreSQL	Relational, Multi-model	550.96	-1.27	+44.02
5.	5.	5.	MongoDB	Document, Multi-model	458.95	+1.73	+25.62
6.	6.	6.	IBM Db2	Relational, Multi-model	157.61	+0.44	-7.94
7.	7.	8.	Redis	Key-value, Multi-model	152.57	-2.44	+1.15
8.	8.	7.	Elasticsearch	Search engine, Multi-model	151.00	-0.25	-1.16
9.	9.	10.	SQLite	Relational	123.17	+1.28	-0.19
10.	10.	11.	Cassandra	Wide column	114.62	-3.46	-5.74
11.	11.	9.	Microsoft Access	Relational	114.17	-1.16	-13.89
12.	12.	13.	MariaDB	Relational, Multi-model	93.89	+0.10	+6.56
13.	13.	12.	Splunk	Search engine	88.54	+0.88	-0.23
14.	16.	14.	Hive	Relational	72.32	+1.89	-11.21
15.	15.	25.	Microsoft Azure SQL Database	Relational, Multi-model	71.29	-0.07	+39.88
16.	14.	15.	Teradata	Relational, Multi-model	70.90	-1.69	-5.91
17.	17.	16.	Amazon DynamoDB	Multi-model	69.14	+0.01	+7.01
18.	18.	21.	SAP Adaptive Server	Relational	52.25	-2.36	-0.48
19.	19.	22.	Neo4j	Graph	52.16	-1.62	+0.96

## DB-Engines Ranking of Cassandra



# Vantagens

- Alta disponibilidade;
- Performance;
- Extremamente tolerante a falhas
- Escalabilidade linear: se o banco atende 100K de requisições, para atender 200K basta dobrar a infraestrutura;
- Sem nenhum ponto único de falha;
- Altamente distribuído;
- Suporta N datacenters nativamente.

# Quando usar?

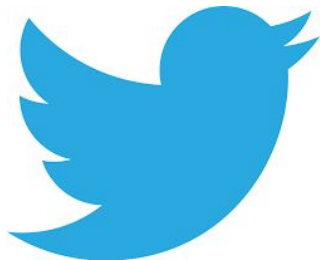
- Se você conhece as suas consultas;
- Se tem um alto volume de dados;
- Se tem mais de 3 servidores;
- Dados desnormalizados;
- Quando você tiver tempo e recursos para modelar os dados;
- Não precisa de integridade referencial;



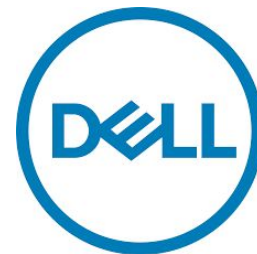
# Quando não utilizar o Cassandra

- Se precisar de muita consistência, a aplicação terá que garantir;
- Se o volume de dados ou o throughput da aplicação for muito pequeno;
- Se o modelo da aplicação não suportar o paradigma colunar;

# Aplicações



# Aplicações





04.

Instalação



## Cassandra Cluster

Container Registry tag: `Cassandra 3.11.x` ▼

[Cassandra Cluster \(Google Click to Deploy containers\)](#)

NoSQL database with masterless replication for high availability

CONFIGURAR



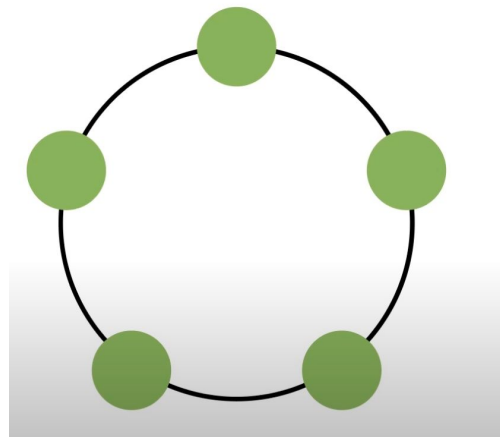
# kubernetes



# Em qual node fica quem?

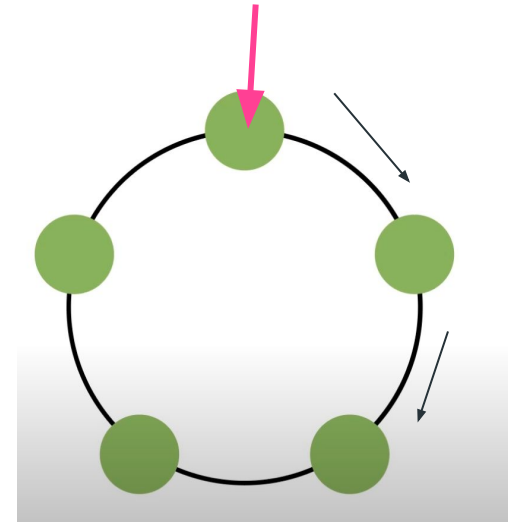
- Tabelas com **Partition Key**
- Passa por uma função hash -> id único (token)
- Divide os tokens pelos nodes

<b>Categoria</b>	<b>Título</b>	<b>Preço</b>	<b>Id</b>
Boneca	Barbie	R\$2000	1
Carrinho	Hot Alguma Coisa	R\$50	2
Boneca	Bratz	R\$1500	3



# Dados duplicados?

- As réplicas garantem que nossos dados estejam disponíveis
- 2 estratégias de replicação: uma **simples** (1 **data center**) e a Network Topology (vários data centers)
- Fator de replicação? 3, cada dado estará em 3 nodes diferentes



# E como eu acesso isso de fora?

- Criamos um service do tipo **External Load Balancer**
- Deveríamos ter colocado usuário e senha
- Mudamos a porta

Name ↑	Status	Type	Endpoints
cassandra-cassandra-svc	✓ OK	Cluster IP	None
cassandra-lb-service	✓ OK	External load balancer	35.196.27.83:7154 ↗
kalm-controller-manager-service	✓ OK	Cluster IP	10.11.250.136
portainer	✓ OK	External load balancer	34.73.183.11:9000 ↗

**External Load Balancer** nos fornece um IP acessível externamente e manda o tráfego pro nosso cluster.





05.



A aplicação  
escolhida





# Dados

Id (uuid)	Stock (counter)	Title (string)
f23acd76-40eb-4472-8234-37b426f5a0b2	72	L.O.L Surprise!
56888270-50ce-4efe-9c80-2c19c4634048	143	Jogo Roda a Roda
123e4567-e89b-12d3-a456-426614174000	7	Barbie Casa Malibu



# Fontes

- <https://medium.com/nstech/apache-cassandra-8250e9f30942>
- [https://docs.datastax.com/en/archived/cassandra/2.0/cassandra/dml/dml\\_config\\_consistency\\_c.html](https://docs.datastax.com/en/archived/cassandra/2.0/cassandra/dml/dml_config_consistency_c.html)
- <http://db4beginners.com/blog/cassandra/>
- <https://www.portainer.io/>
- <https://www.devmedia.com.br/como-usar-o-apache-cassandra-em-aplicacoes-java-ee-parte-1/34409>

# Fontes

- [https://db-engines.com/en/ranking\\_trend/system/Cassandra](https://db-engines.com/en/ranking_trend/system/Cassandra)
- <https://stack.g2.com/apache-cassandra>
- <https://dcomp.ufscar.br/verdi/topicosCloud/Cassandra.pdf>
- <https://www.datastax.com/node/1162>
- <https://www.datastax.com/node/1162>
- <https://docs.datastax.com/en/cassandra-oss/3.0/cassandra/architecture/archDataDistributeFailDetect.html>
- <https://www.onsip.com/voip-resources/voip-fundamentals/intro-to-cassandra-and-networktopologystategy>