NoSQL Apache Cassandra para DBAs

Conceitos básicos que todo DBA deve conhecer sobre Apache Cassandra.

DBA BRASIL 2.0

IBTA
Instituto Brasileiro de Tecnologia Avancada

Nossos Patrocinadores

DELL EMC

TmaxSoft Brasil

STROHL Brasil

Timbira
A empresa brasileira de PostgreSQL

GREEN tecnologia

DBA 4 All
We care about your data

DB Academy

ORAMASTER

Apresentação Pessoal

- **Ronaldo Martins:** Há mais de 14 anos dedicado à tecnologias Oracle, passando pelas releases 8i, 9i, 10g, 11g e 12c do Oracle Database. Consultor DBA Oracle, MySQL, NoSQL Apache Cassandra, instrutor de treinamentos Oracle e fundador da TIMEDATA Tecnologia.

- Blog: http://ronaldolmartins.blogspot.com.br/

Tópicos

- Breve História do Apache Cassandra;
- Apache Cassandra x Modelo Relacional;
- O que o Apache Cassandra não tem?
- Principais componentes do Apache Cassandra;
- Entendendo Escrita, Leitura no Apache Cassandra;

Tópicos

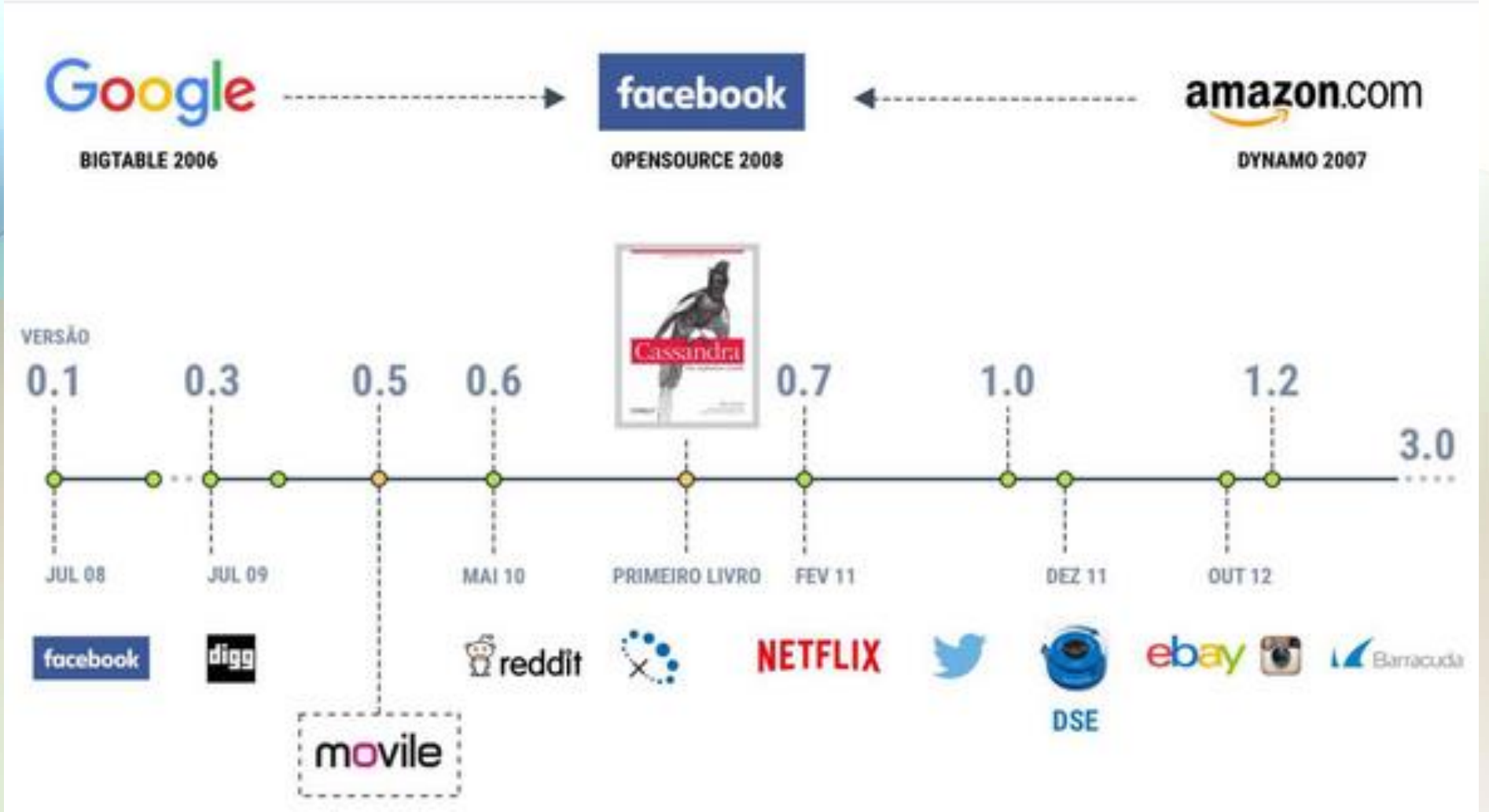
- Backup;
- Escalabilidade;
- Cluster;
- Replication Factor;
- Consistência de dados;

Sobre Cassandra

Sobre Apache Cassandra

Apache Cassandra é um **banco de dados** não relacional (**NoSQL**) orientado a colunas, distribuído, escalável, de alta disponibilidade, tolerante a falhas.

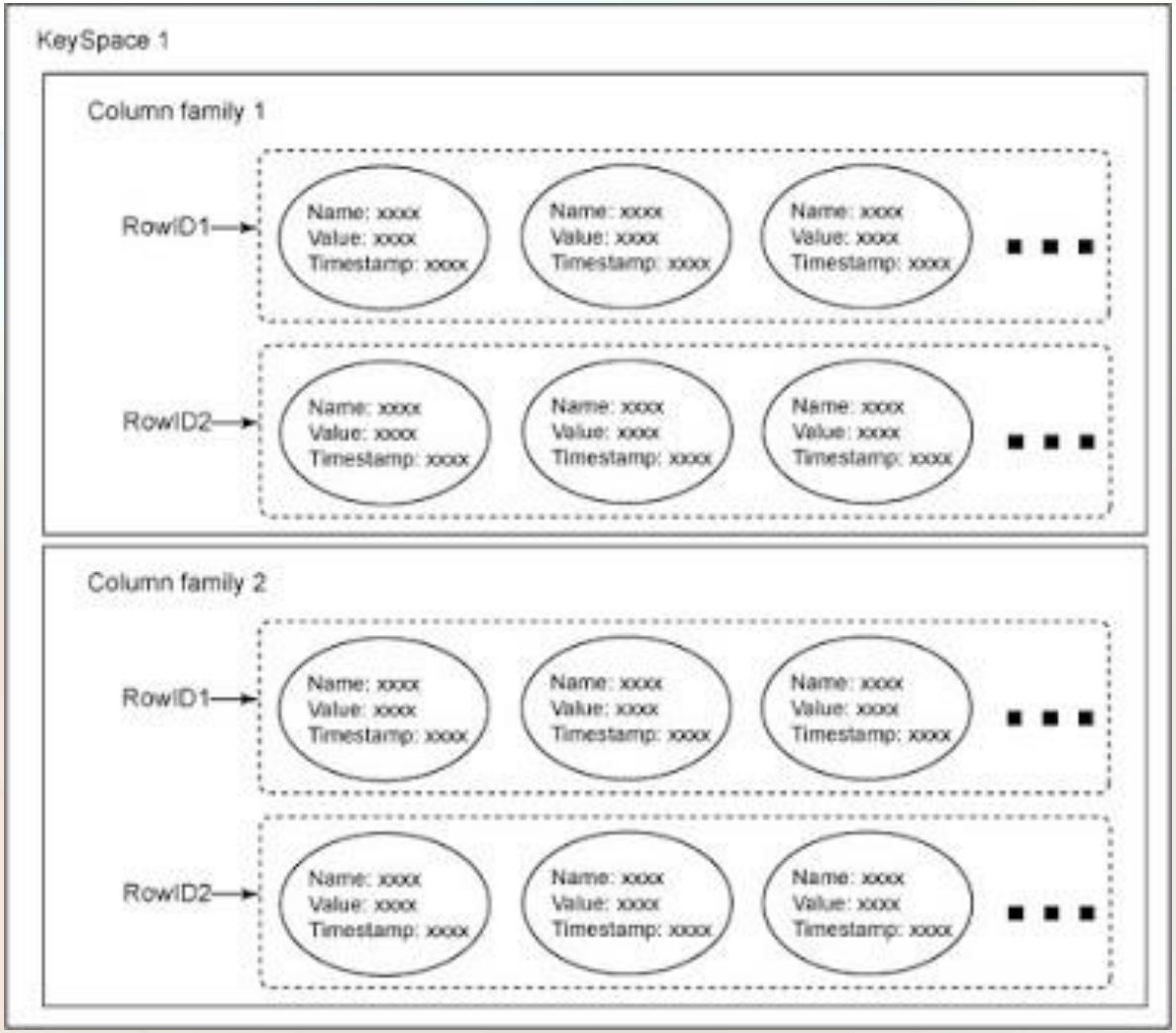
Sobre Apache Cassandra



Apache Cassandra x Modelo Relacional

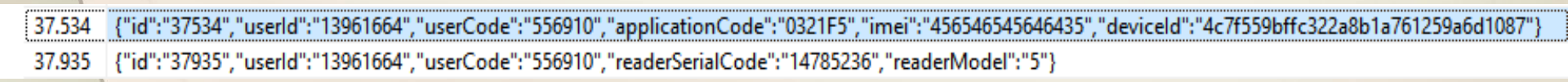
Relational Model	Cassandra Model
Database	Keyspace
Table	Column Family (CF)
Primary key	Row key
Column name	Column name/key
Column value	Column value

Apache Cassandra x Modelo Relacional – Modelo de dados Chave/Valor



Modelo de dados Flexível

- Um “alter table em ambiente OLTP” x “Modificação de coluna no Cassandra”



Linguagem CQL

- Cassandra Query Language (CQL);
- O uso do CQL é semelhante ao SQL;
- Utilitário de linha de comando **cqlsh** ou DevCenter;

Time-To-Live - TTL

- INSERT INTO TB01 (id,json) VALUES (99999,'teste') **using TTL 20;**

- select **TTL(**json**),** id, json from TB01;

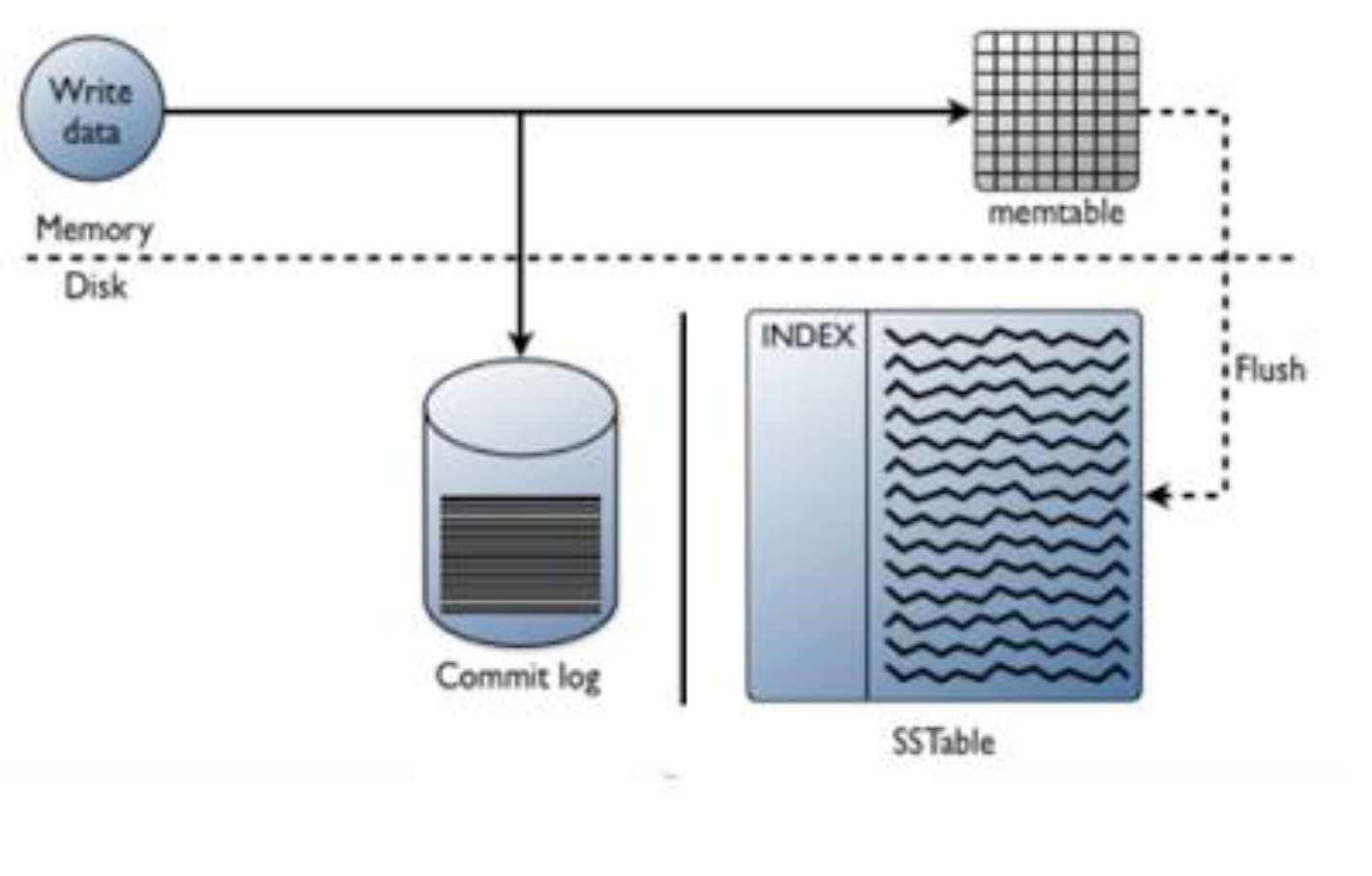
Tombstones

- O Cassandra realiza uma marca nos dados TTL depois que a quantidade de tempo solicitada expirou. Depois que os dados são marcados, os dados são removidos automaticamente durante o processo normal de compactação;

O que o Apache Cassandra não tem?

- Subconsultas;
- JOINs (máximo de dados possível na mesma linha);
- Chaves estrangeiras;

Principais componentes do Apache Cassandra



Principais componentes do Apache Cassandra

- **Commit log** - Mecanismo de recuperação. Operação de escrita é registrada no registro de commit. É sequencial;
- **Mem-table** - É uma estrutura de dados residente em memória. Após commit log, os dados serão nela;

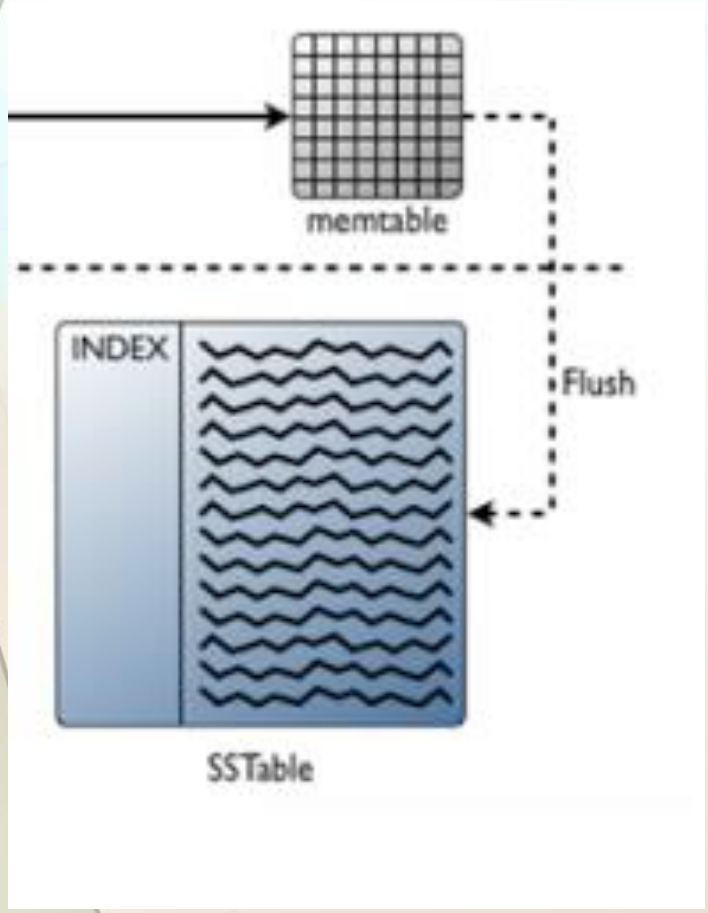
Principais componentes do Apache Cassandra

- **SSTable** - Arquivo em disco para o qual os dados são liberados a partir da mem-table ou quando o seu conteúdo chegar a um valor limite;
- **Filtro de Bloom** - É um tipo especial de cache. Filtros Bloom são acessados depois de cada consulta;

Entendendo Escrita no Apache Cassandra

- Escreve primeiro em um registro “log”;
- Depois escreve em uma estrutura de tabela na memória denominada por mem-table.

Entendendo Escrita no Apache Cassandra

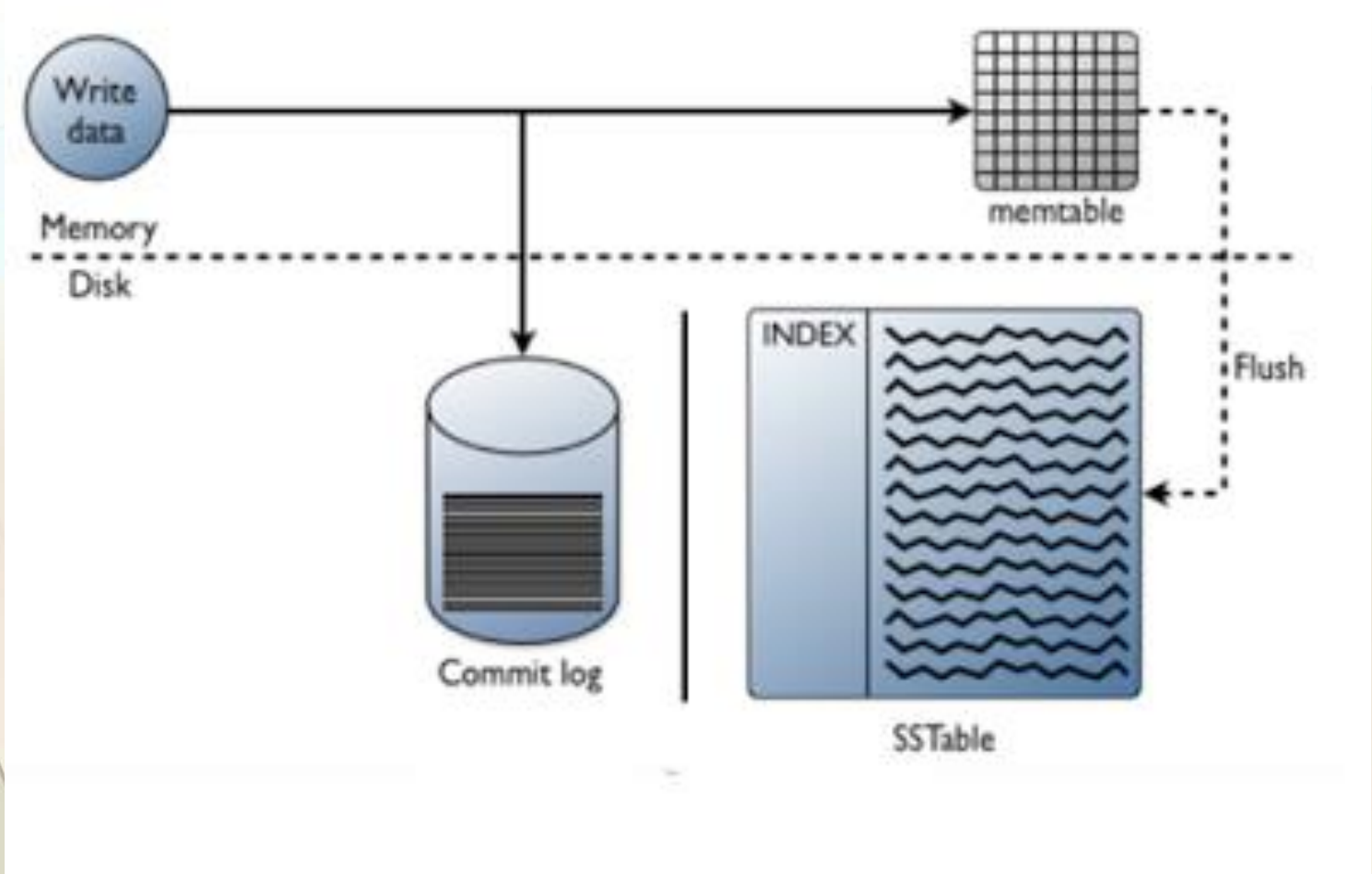


- Parâmetro: **memtable_flush_queue_size;**
- Escritas são agrupadas na memória e periodicamente gravadas em disco;

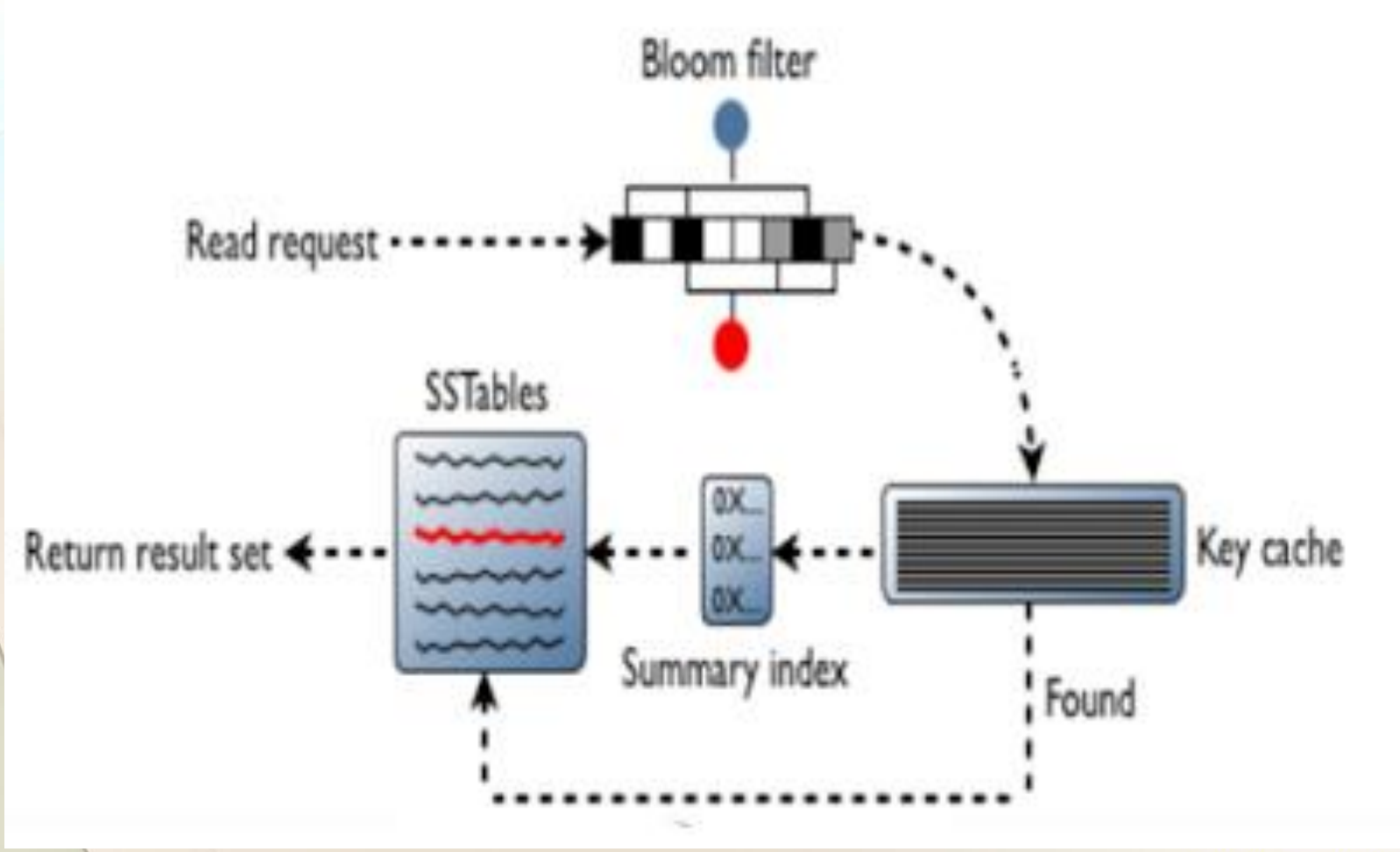
Entendendo Escrita no Apache Cassandra

- A escrita é bem-sucedida uma vez que é escrita no log e na memória(Mem-table), gerando o mínimo de I/O no disco no momento de gravação;
- Se ocorrer uma queda ou falha do servidor antes da mem-table ser flushed, o commit log é repetido ao reiniciar para recuperar escritas perdidas;

Entendendo Escrita no Apache Cassandra



Entendendo Leitura no Apache Cassandra



Entendendo Leitura no Apache Cassandra

- O Key cache salva uma busca durante a operação de leitura. Se uma chave não for encontrada no key cache, o Summary Index será pesquisado.
- O Summary Index é uma estrutura em memória que armazena uma amostragem do índice e mapeia o local de localização de cada chave no arquivo ou bloco que tem os dados no SSTable.

Entendendo Leitura no Apache Cassandra

- SSTable tem um filtro Bloom (estrutura em memória) associado a ele que verifica se uma chave de linha solicitada existe no SSTable antes de fazer qualquer busca no disco;
- Em seguida, o Cassandra checa o key cache global(localização das chaves na memória). Se os dados solicitados não estão no key cache, é feita uma busca binária no index summary para encontrar uma linha.

Entendendo Leitura no Apache Cassandra

- Finalmente, o Cassandra realiza um simples seek e uma leitura sequencial de colunas no SSTable, e retorna o conjunto de resultados.

Backup

- Snapshot de todos os arquivos de dados no disco (arquivos SSTable);

```
$ nodetool -h localhost -p 7199 snapshot mykeyspace
```

Escalabilidade

- O Cassandra trabalha na forma horizontal ou linear que é a capacidade de adicionar novas máquinas no cluster para, de forma distribuída, aumentar os recursos de processamento, memória e disco.

Cluster

- Sistema distribuído peer-to-peer através de seus nós, e os dados são distribuídos entre todos os nós em um cluster, são todos iguais.
  - Todos os nós em um cluster desempenham o mesmo papel. Cada nó é independente e, ao mesmo tempo interligados;

Cluster

- Cada nó em um cluster pode aceitar leitura e escrita pedidos, independentemente de onde os dados são realmente localizado no cluster;

Cluster - Configuração

- /etc/cassandra/cassandra.yaml.

```
cluster_name: 'CassandraClusterTest'
num_tokens: 256
seed_provider:
- class_name: org.apache.cassandra.locator.SimpleSeedProvider
  parameters:
    - seeds: "cass01,cass02"

listen_address: 192.168.0.105
rpc_address: 0.0.0.0
broadcast_rpc_address: 192.168.0.105
endpoint_snitch: GossipingPropertyFileSnitch
auto_bootstrap: false
```

Replication Factor

- O Cassandra armazena réplicas em vários nós para garantir confiabilidade e tolerância a falhas.
- Um fator de replicação de 2 significa duas cópias de cada linha, onde cada cópia está em um nó diferente.
- O fator de replicação não deve exceder o número de nós no cluster.

Replication Factor

- ```
  create keyspace TestKeyspace
  WITH REPLICATION = { 'class' :
  'SimpleStrategy',
  'replication_factor' : '2' };
  ```
  ```

Consistência de dados

- Níveis de consistência na Cassandra podem ser configurados para leitura/gravação para gerenciar a disponibilidade versus a precisão dos dados.

Consistência de dados - Escrita

- O nível de consistência determina o número de réplicas nas quais a gravação deve ser bem-sucedida antes de retornar uma confirmação cliente.

Consistência de dados - Escrita

- **ONE**: Uma gravação deve acontecer em pelo menos um nó de réplica, antes da confirmação para o usuário.(Commit-Log e Mem-table)
- **TWO**: Uma gravação deve acontecer em pelo menos dois nós de réplica, antes da confirmação para o usuário.

Consistência de dados - Leitura

- O nível de consistência também especifica quantas réplicas devem responder a uma solicitação de leitura antes de retornar dados para o aplicativo cliente.

Consistência de dados - Leitura

- **TWO:** Retorna os dados mais recentes de duas das réplicas mais próximas. O Cassandra escolhe pelo timestamp o dado mais recente entre as réplicas que responderam;

Consistência de dados - Leitura

37.534	{"id":"37534","userId":"13961664","userCode":"556910","applicationCode":"0321F5","imei":"456546545646435","deviceId":"4c7f559bffc322a8b1a761259a6d1087"}
37.935	{"id":"37935","userId":"13961664","userCode":"556910","readerSerialCode":"14785236","readerModel":"5"}

Details

Type: varchar

Timestamp: 1487079773564000
2017-02-14 13:42:53+0000

TTL: null

Value: {"id":"37534","userId":"13961664","userCode":"556910","applicationCode":"0321F5","imei":"456546545646435","deviceId":"4c7f559bffc322a8b1a761259a6d1087"}

Details

Type: varchar

Timestamp: 1488463135652000
2017-03-02 13:58:55+0000

TTL: null

Value: {"id":"37935","userId":"13961664","userCode":"556910","readerSerialCode":"14785236","readerModel":"5"}

Consistência de dados - CQL

- `CONSISTENCY TWO;`
- `INSERT INTO Alunos(nome,matrícula) VALUES ('FFFF', 65535) ;`

Consistência de dados - CQL

- `CONSISTENCY THREE;`
- `SELECT * FROM Alunos where matricula = 999 ;`

Perguntas?

- **Blog**: http://ronaldolmartins.blogspot.com.br/
- **Email**: ronaldolmartins@gmail.com

DBA BRASIL 2.0

IBTA
Instituto Brasileiro de Tecnologia Avancada

Nossos Patrocinadores

DELL EMC

TmaxSoft Brasil

STROHL Brasil

Timbira
A empresa brasileira de PostgreSQL

GREEN tecnologia

DBA 4 All
We care about your data

DB Academy

ORAMASTER