Week 7: Column-family Stores

DIANE WOODBRIDGE, PH.D

Cassandra Interview Questions

Cassandra data model

Tunable consistency (Quorum)

Difference between RDBMS and Cassandra

CAP Theorem

Cassandra query language

cqlsh

Compaction

Super column

Column family

Memtable

SSTable

Keyspace

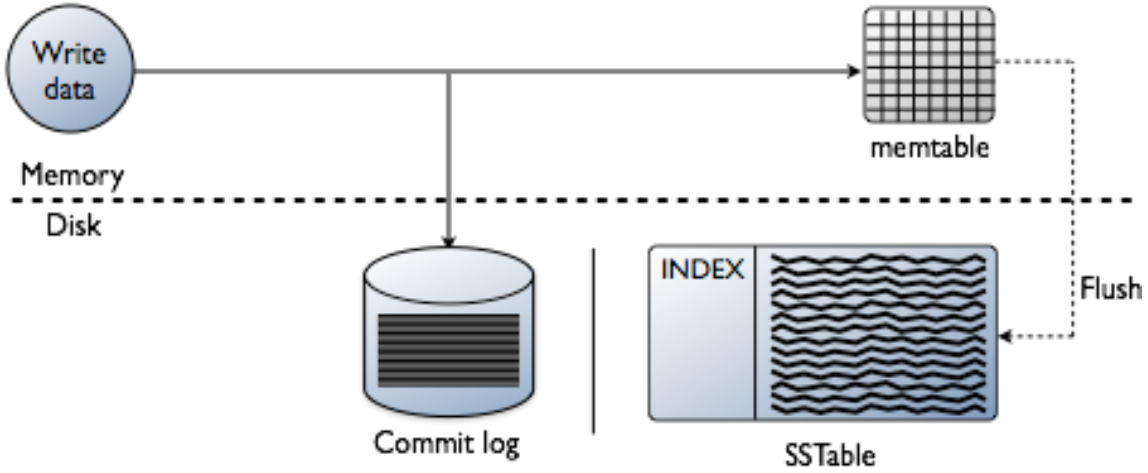
Tombstone

Cassandra

Writes

VERY FAST! (WHY??)

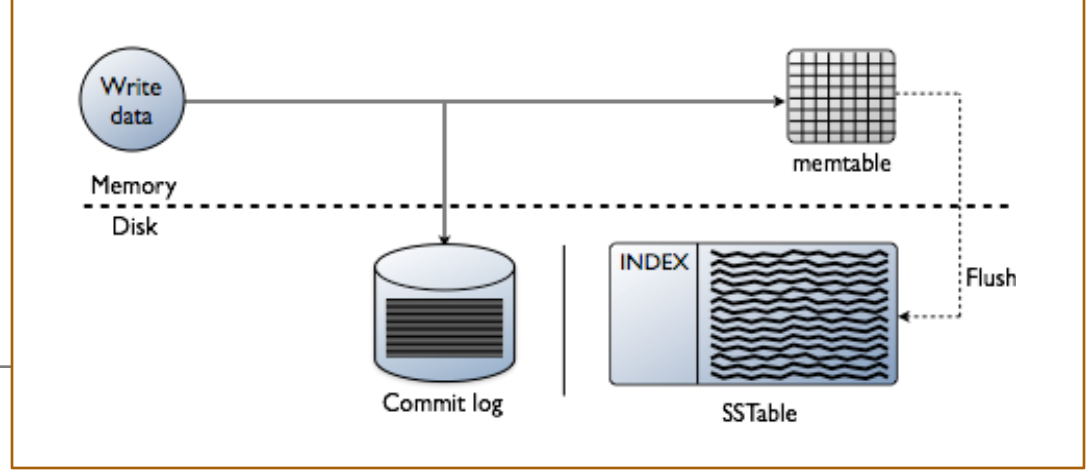
- When a write is received by Cassandra, the data is first recorded in a **commit log**, then written to an in-memory structure called **memtable**. A write operation is considered successful once it is written to the commit log and the memtable. Writes are batched in memory and periodically written out to structures known as **SSTable**.



1. Logging data in the commit log
2. Writing data to the memtable
3. Flushing data from the memtable
4. Storing data on disk in SSTables
5. Compaction

https://docs.datastax.com/en/cassandra/2.2/cassandra/dml/dmlHowDataWritten.html

Cassandra



Writes

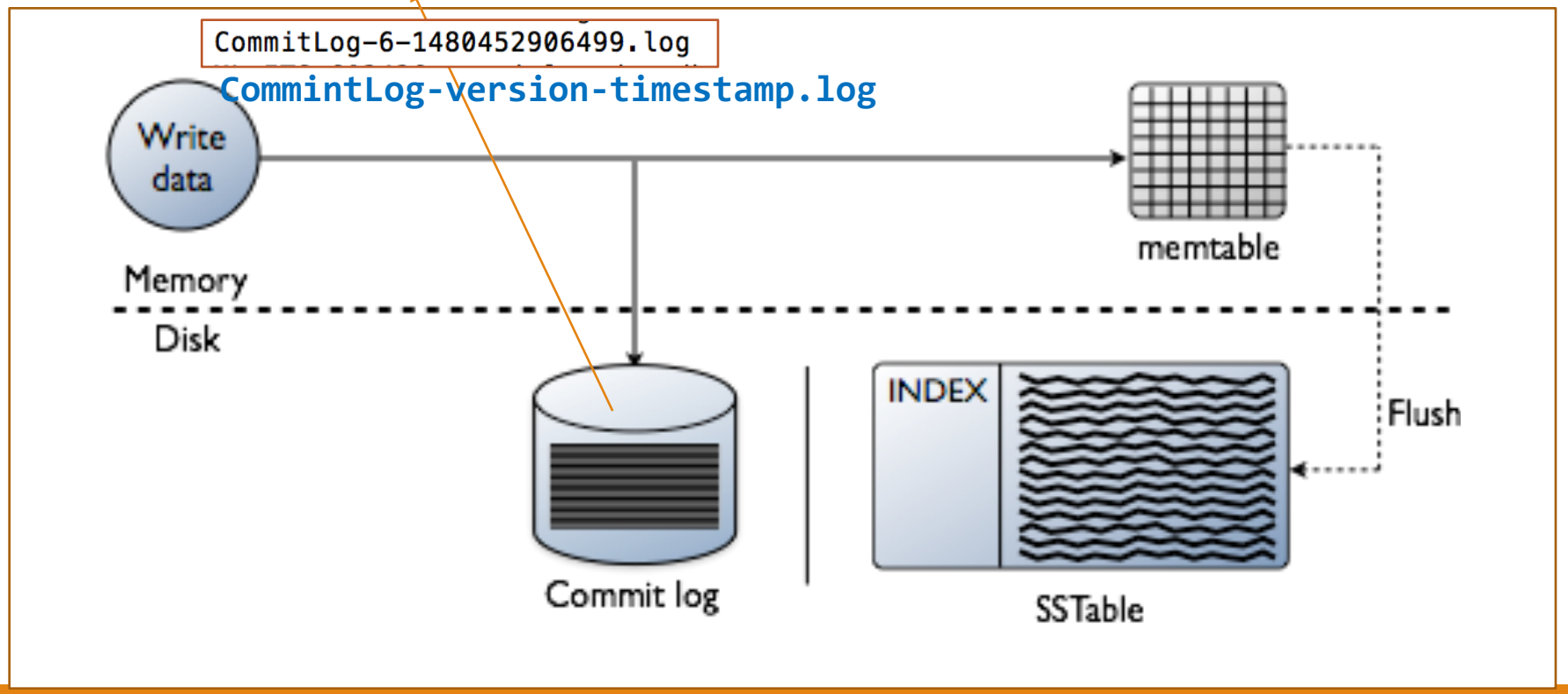
1. **Logging data in the commit log.** : It is a crash-recovery mechanism. If a write operation does not make it to the in-memory store (memtable), it will still be possible to recover the data.
2. **Writing data to the memtable.** : memtable is a memory-resident data structure which stores data rows that can be looked up by key.
3. **Flushing data from the memtable.** : When he number of objects stored in the memtable reaches a threshold, the contents are flushed to disk called SSTable .
4. **Storing data on disk in SSTable.** : SSTables are immutable, not written to again after the memtable is flushed.
5. **Periodic compaction.** : Reorganize SSTables for better future read performance.

https://docs.datastax.com/en/cassandra/2.2/cassandra/dml/dmlHowDataWritten.html

Cassandra

Writes

```
ML-ITS-603436:~ dwoodbridge$ cd /usr/local/var/lib/cassandra/
ML-ITS-603436:cassandra dwoodbridge$ ls
commitlog       data            hints           saved_caches
```



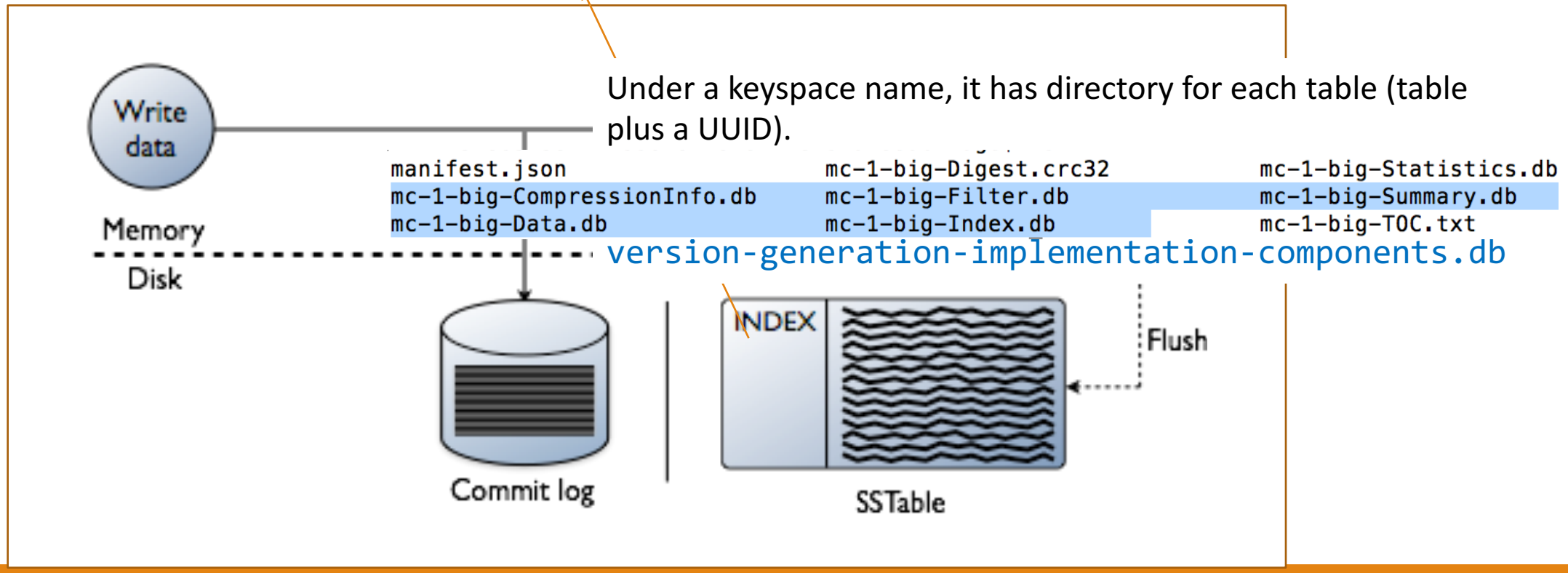
https://docs.datastax.com/en/cassandra/2.2/cassandra/dml/dmlHowDataWritten.html

Cassandra

Writes

```
ML-ITS-603436:~ dwoodbridge$ cd /usr/local/var/lib/cassandra/
ML-ITS-603436:cassandra dwoodbridge$ ls
commitlog       data            hints           saved_caches
```

Under a keyspace name, it has directory for each table (table plus a UUID).

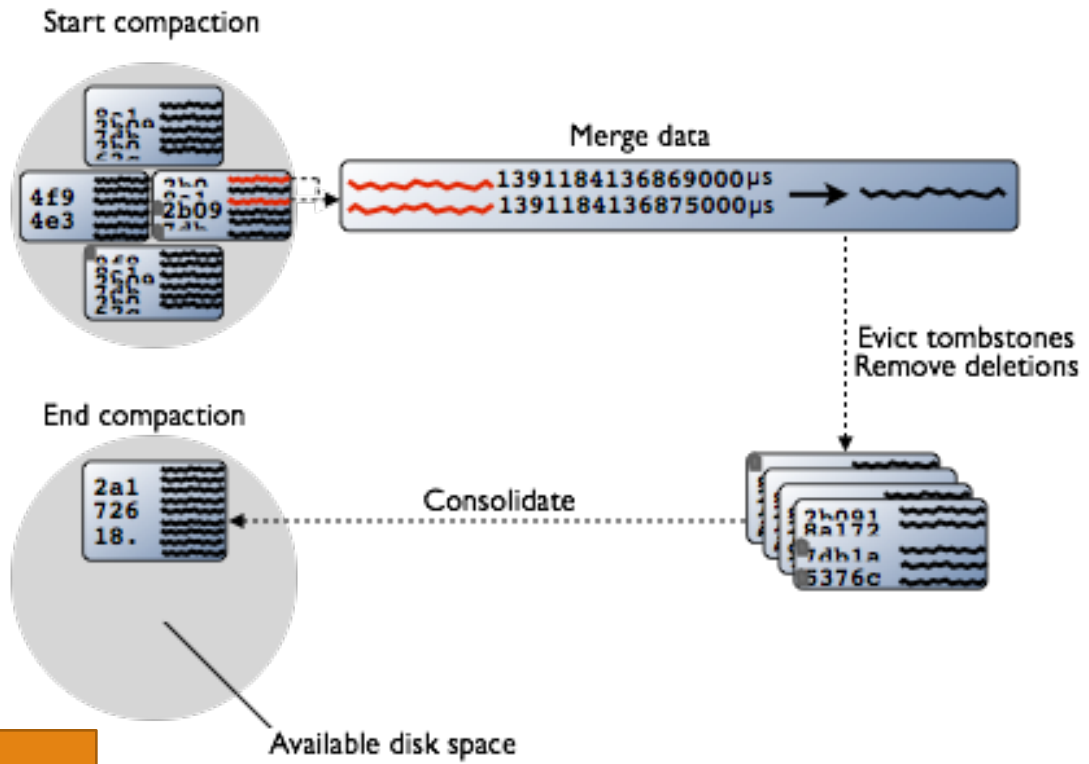


https://docs.datastax.com/en/cassandra/2.2/cassandra/dml/dmlHowDataWritten.html

Cassandra

Writes

- Compaction in SSTables
  - Periodically clean out stale data values to support fast read performance.
    - Merge SSTables. : The keys are merged, columns are combined, tombstones are discarded, and a new index is created to a new SSTable.

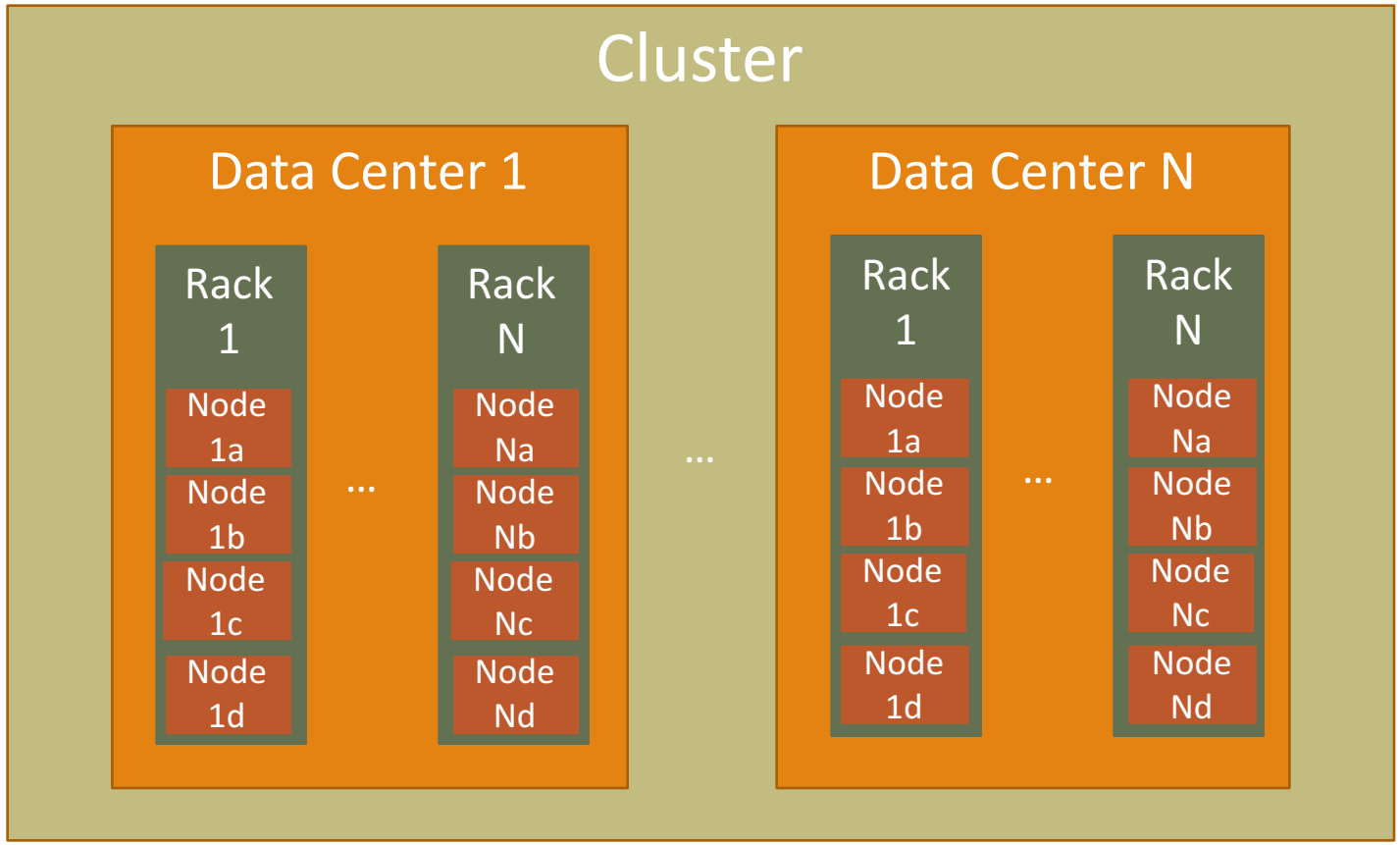


Tombstones (Cassandra Deletes)
When you execute a delete operation, the data is not immediately deleted. Instead, it is treated as an update operation that places a tombstone on the value for GCGraceSeconds for eventual consistency.

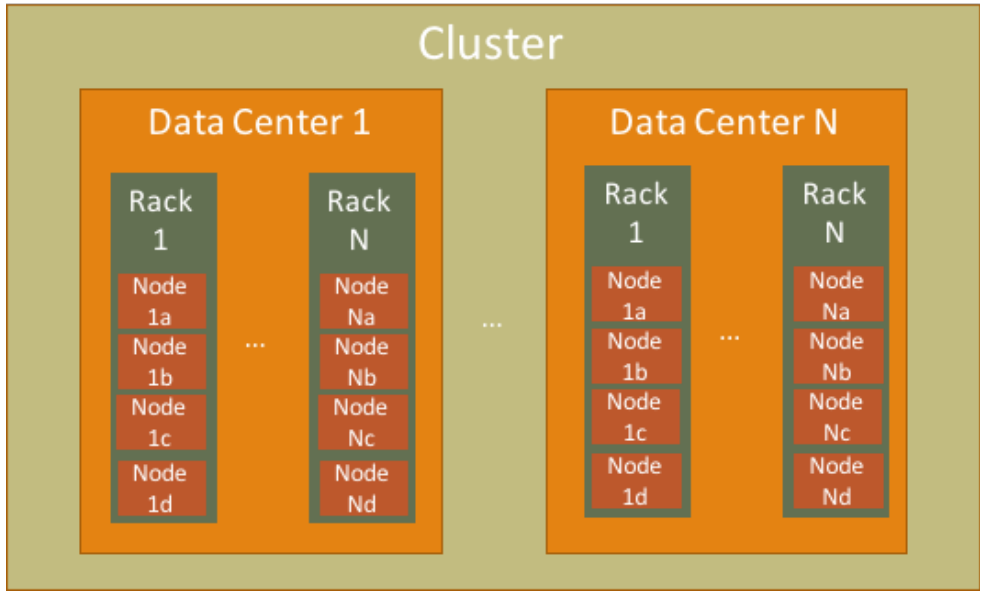
Cassandra Reads :
https://docs.datastax.com/en/cassandra/3.x/cassandra/dml/dmlAboutReads.html

https://docs.datastax.com/en/cassandra/2.1/cassandra/dml/dml_write_path_c.html#concept_ds_wt3_32w_zj__dml-compaction
https://docs.datastax.com/en/cassandra/2.0/cassandra/dml/dml_about_deletes_c.html

Cluster Topology in Cassandra



Cluster Topology in Cassandra



Node (vnode): A node is the storage layer within a server.

Rack : A logical set of nodes in close proximicity .

Data Center : A logical set of racks.

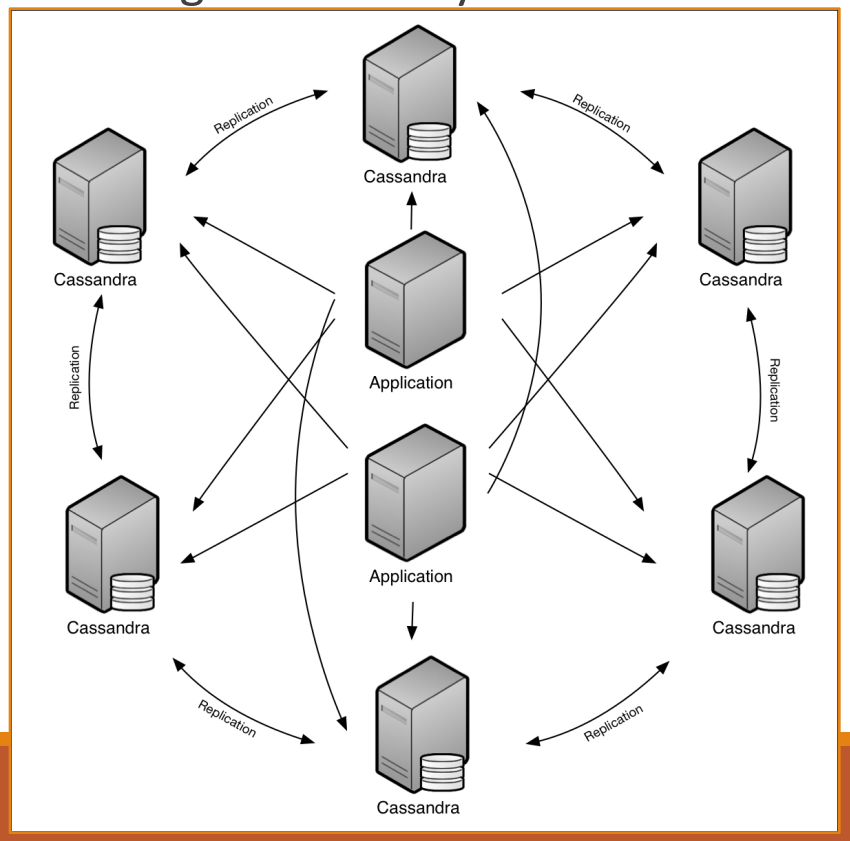
Cluster : A set of data centers.

Snitches : Gather information about the topology so that Cassandra can efficiently route requests.

Cassandra

Replication

- All replicas are equally important; there is no primary or master replica.
- Uses peer-to-peer replication. ➔ High availability.



Cassandra

Replication Strategies (2)

- SimpleStrategy
  - Use only for a single data center.
  - Not aware of their placement on a data center rack
- NetworkTopologyStrategy
  - Use NetworkTopologyStrategy when you have (or plan to have) your cluster deployed across multiple data centers. This strategy specify how many replicas you want in each data center. NetworkTopologyStrategy attempts to place replicas on distinct racks because nodes in the same rack (or similar physical grouping) often fail at the same time due to power, cooling, or network issues.
  - Much easier to expand to multiple data centers when required by future expansion.

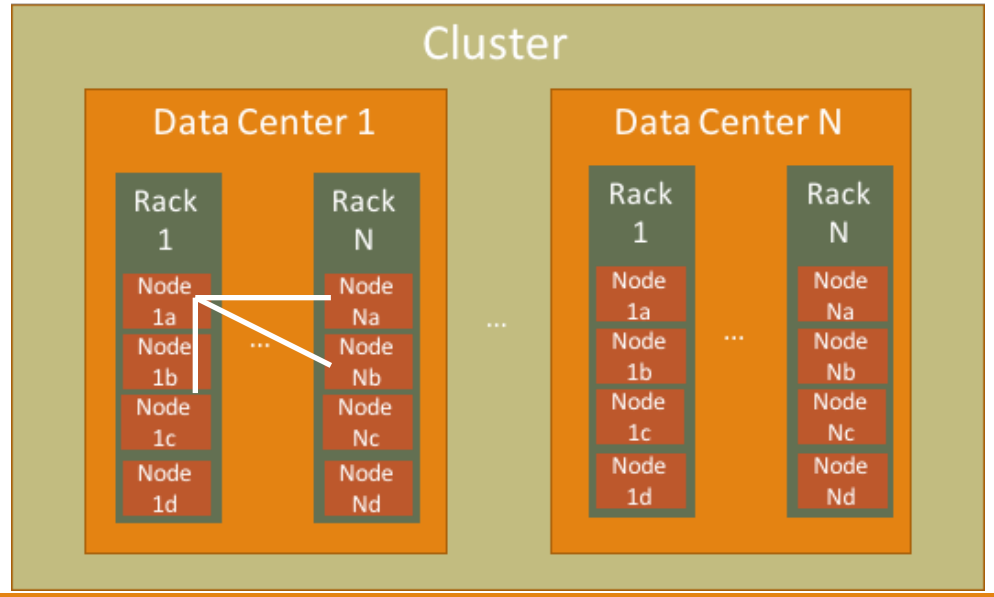
https://docs.datastax.com/en/cassandra/2.0/cassandra/architecture/architectureDataDistributeReplication_c.html

Cassandra

Replication Strategies (2)

- SimpleStrategy
  - Syntax : { 'class' : 'SimpleStrategy', 'replication_factor' : <integer> };
  - Ex.

```
cqlsh> CREATE KEYSPACE mydb WITH REPLICATION = { 'class' : 'SimpleStrategy', 'replication_factor' : 3 };
```



https://docs.datastax.com/en/cql/3.1/cql/cql_reference/create_keyspace_r.html

Cassandra

Replication Strategies (2)

- NetworkTopologyStrategy
  - Syntax : { 'class' : 'NetworkTopologyStrategy'[, '<data center>' : <integer>, '<data center>' : <integer>] . . . };
  - Ex.

To determine the default data center name, use nodetool status.

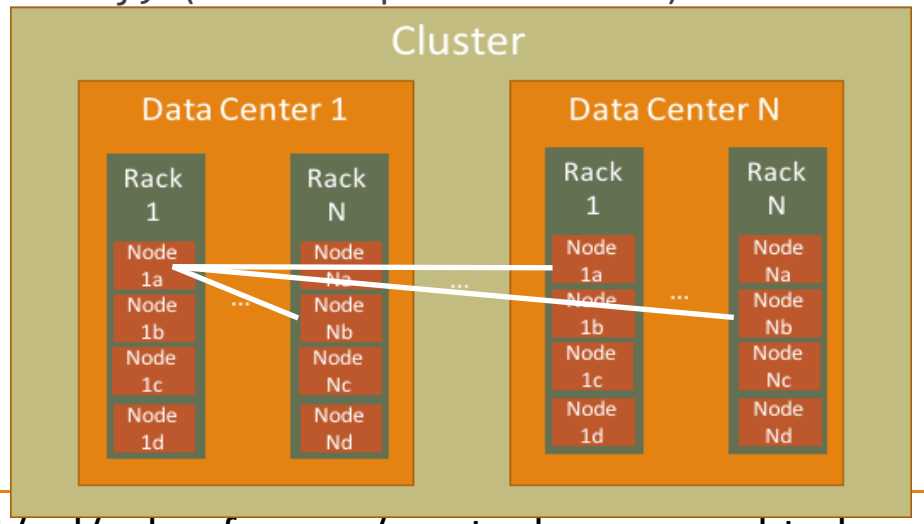
```
ML-ITS-603436:~ dwoodbridge$ nodetool status
objc[28141]: Class JavaLaunchHelper is implemented in both /Library/Java/JavaVirtualMachines/jdk1.8.0_101.jdk/Contents/Home/bin/java and /Library/Java/JavaVirtualMachines/
jdk1.8.0_101.jdk/Contents/Home/jre/lib/libinstrument.dylib. One of the two will be used. Which one is undefined.
Datacenter: datacenter1
=======================
Status=Up/Down
|/ State=Normal/Leaving/Joining/Moving
--  Address    Load       Tokens       Owns    Host ID                               Rack
UN  127.0.0.1  578.8 KiB  256          ?       289235b7-950c-42d0-891a-22e8fa4357e2  rack1

Note: Non-system keyspaces don't have the same replication settings, effective ownership information is meaningless
```

https://docs.datastax.com/en/cql/3.1/cql/cql_reference/create_keyspace_r.html

Cassandra

Replication Strategies (2)

- NetworkTopologyStrategy
  - Syntax : { 'class' : 'NetworkTopologyStrategy'[, '<data center>' : <integer>, '<data center>' : <integer>] . . . };
    - `cqlsh> CREATE KEYSPACE mydb2 WITH REPLICATION = { 'class' : 'NetworkTopologyStrategy', 'datacenter1' : 3 };`
    - `CREATE KEYSPACE mydb WITH REPLICATION = {'class' : 'NetworkTopologyStrategy', 'dc1' : 3, 'dc2' : 2};` (With multiple data centers.)

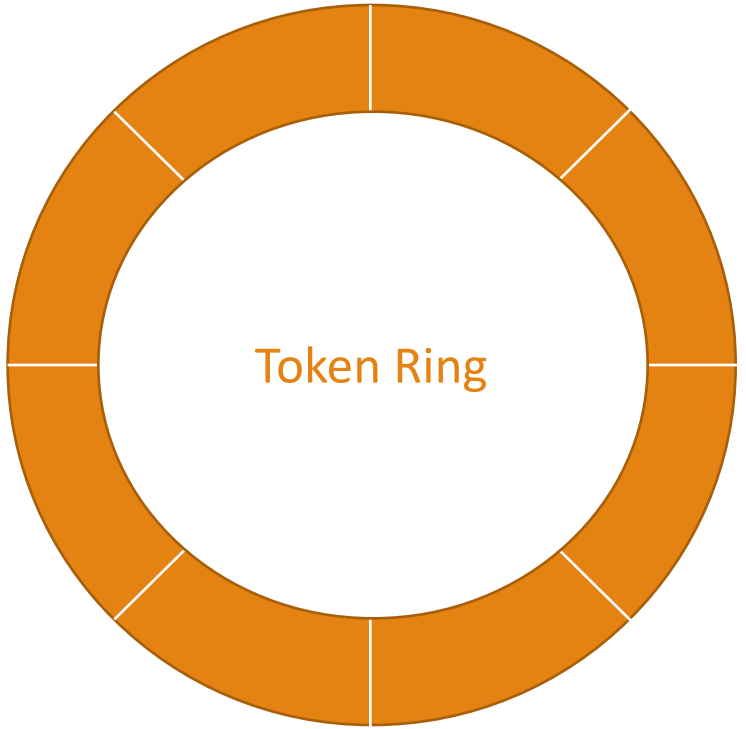


https://docs.datastax.com/en/cql/3.1/cql/cql_reference/create_keyspace_r.html

Cassandra

Partitioning (Sharding)

- Token Ring
  - Each node is assigned to a position in a ring with a range of the token values.
    - Token : 64-bit integer ID used to identify each partition.
  - Try to arrange consecutive token ranges to be spread across nodes in different racks.
- Partitioner
  - Determines how data is distributed across the nodes in the cluster by computing the partition key token.
  - By default, it uses hash function. (Murmur3Partitioner)

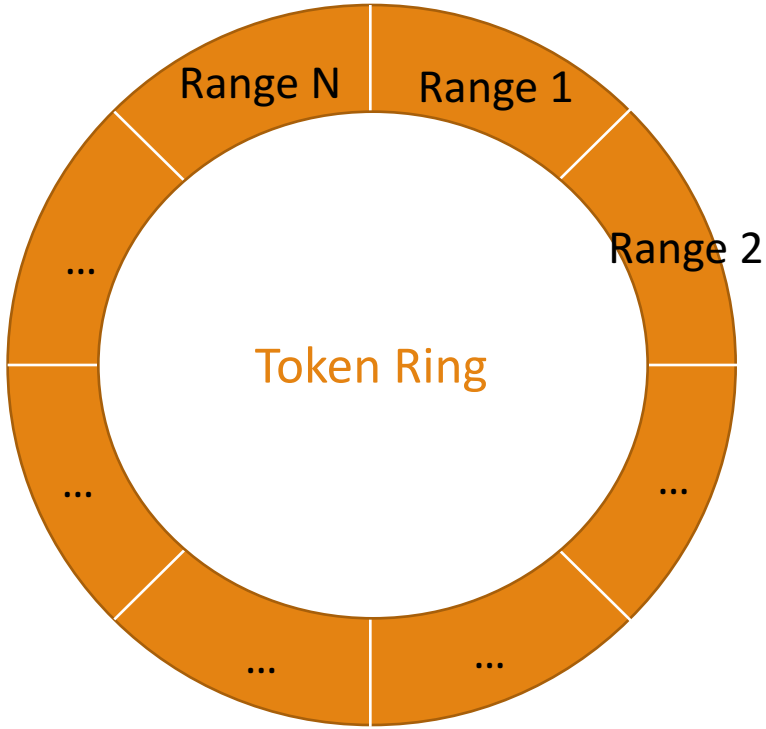


https://docs.datastax.com/en/cassandra/2.0/cassandra/architecture/architecturePartitionerAbout_c.html
https://docs.datastax.com/en/datastax_enterprise/4.5/datastax_enterprise/deploy/deployCalcTokens.html

Cassandra

Partitioning (Sharding)

- Token Ring
  - Each node is assigned to a position in a ring with a range of the token values.
    - Token : 64-bit integer ID used to identify each partition.
  - Try to arrange consecutive token ranges to be spread across nodes in different racks.
- Partitioner
  - Determines how data is distributed across the nodes in the cluster by computing the partition key token.
  - By default, it uses hash function. (Murmur3Partitioner)

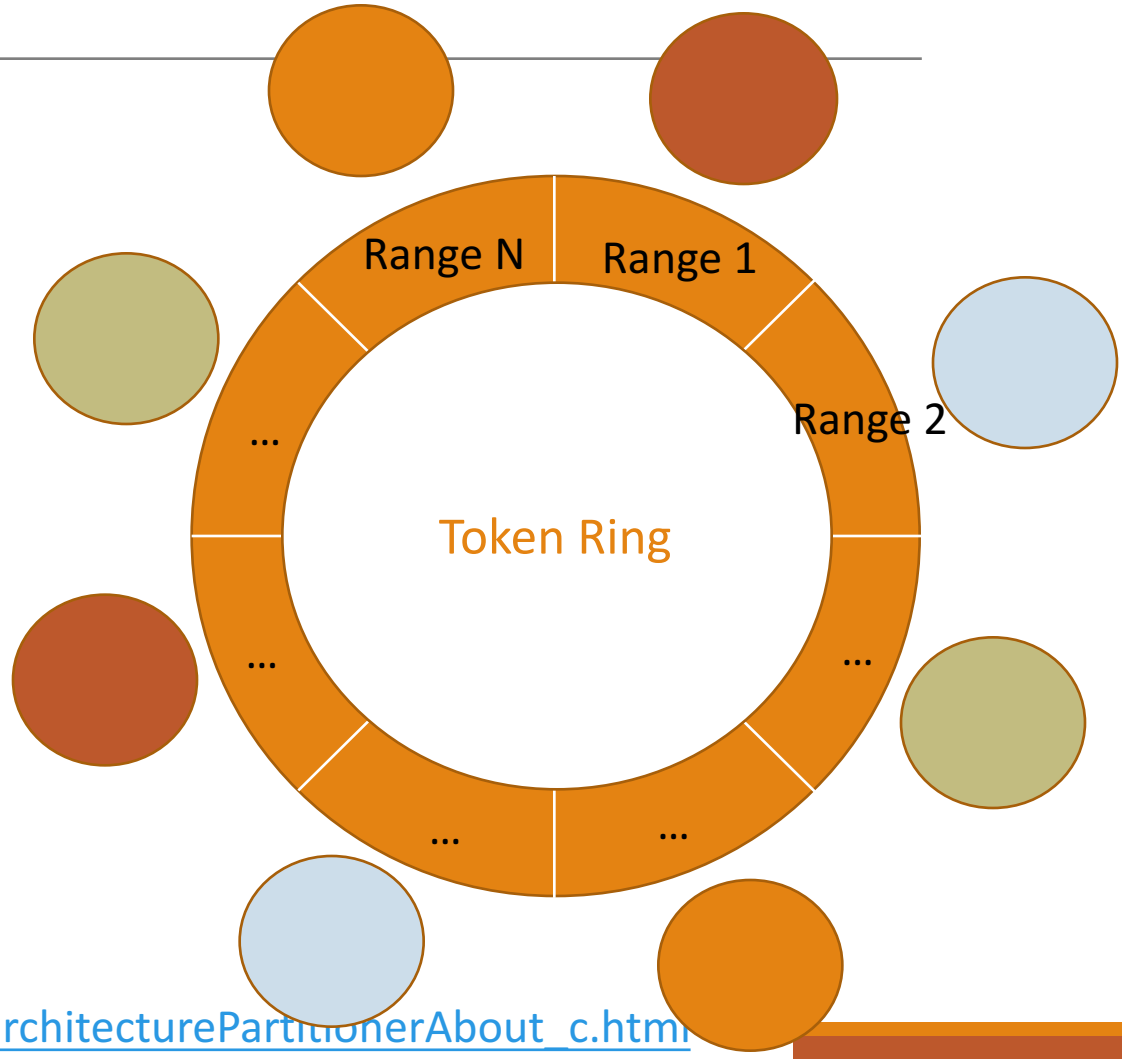


https://docs.datastax.com/en/cassandra/2.0/cassandra/architecture/architecturePartitionerAbout_c.html
https://docs.datastax.com/en/datastax_enterprise/4.5/datastax_enterprise/deploy/deployCalcTokens.html

Cassandra

Partitioning (Sharding)

- Token Ring
  - Each node is assigned to a position in a ring with a range of the token values.
    - Token : 64-bit integer ID used to identify each partition.
  - Try to arrange consecutive token ranges to be spread across nodes in different racks.
- Partitioner
  - Determines how data is distributed across the nodes in the cluster by computing the partition key token.
  - By default, it uses hash function. (Murmur3Partitioner)



https://docs.datastax.com/en/cassandra/2.0/cassandra/architecture/architecturePartitionerAbout_c.html
https://docs.datastax.com/en/datastax_enterprise/4.5/datastax_enterprise/deploy/deployCalcTokens.html

Cassandra

And also install pyYaml.

Creating a cluster.

- CCM (Cassandra Cluster Manager) : Run a multi-node cluster on a single machine.

```
ML-ITS-603436:~ dwoodbridge$ pip install ccm
Collecting ccm
  Downloading ccm-2.4.6.tar.gz (66kB)
    100% |████████████████████████████████| 71kB 979kB/s
Requirement already satisfied: pyYaml in ./anaconda/lib/python2.7/site-packages (from ccm)
Requirement already satisfied: six>=1.4.1 in ./anaconda/lib/python2.7/site-packages (from ccm)
Building wheels for collected packages: ccm
  Running setup.py bdist_wheel for ccm ... done
  Stored in directory: /Users/dwoodbridge/Library/Caches/pip/wheels/fe/a5/76/737b9fd3863af65a59f8958f656f74e0035b919a91db0e1027
Successfully built ccm
Installing collected packages: ccm
Successfully installed ccm-2.4.6
ML-ITS-603436:~ dwoodbridge$ ccm
Missing arguments
Usage:
  ccm <cluster_cmd> [options]
  ccm <node_name> <node_cmd> [options]

Where <cluster_cmd> is one of
  create         Create a new cluster
  add            Add a new node to the current cluster
  populate       Add a group of new nodes with default options
  list           List existing clusters
  switch         Switch of current (active) cluster
  status         Display status on the current cluster
```

Cassandra

Creating a cluster.

- `ccm create -v cassandra_version -n number_of_nodes_for_the_cluster cluster_name`

```
ML-ITS-603436:~ dwoodbridge$ ccm create -v 3.9 -n 3 test
Current cluster is now: test

ML-ITS-603436:~ dwoodbridge$ ccm list
 *test

ML-ITS-603436:~ dwoodbridge$ ccm status
Cluster: 'test'
---------------
node1: DOWN (Not initialized)
node3: DOWN (Not initialized)
node2: DOWN (Not initialized)
```

https://academy.datastax.com/planet-cassandra/getting-started-with-ccm-cassandra-cluster-manager

Cassandra

Starting a cluster.

- If you're running on Mac OSX, create a new interface for every node besides the first, for example if you populated your cluster with 3 nodes, create interfaces for 127.0.0.2 and 127.0.0.3 like so.

```
ML-ITS-603436:~ dwoodbridge$ sudo ifconfig lo0 alias 127.0.0.2
Password:
ML-ITS-603436:~ dwoodbridge$ sudo ifconfig lo0 alias 127.0.0.3
```


https://academy.datastax.com/planet-cassandra/getting-started-with-ccm-cassandra-cluster-manager

Cassandra

Starting a cluster.

- `ccm start`

```
ML-ITS-603436:~ dwoodbridge$ ccm start
[node1 ERROR] objc[1277]: Class JavaLaunchHelper is implemented in both /Libr
ary/Java/JavaVirtualMachines/jdk1.8.0_101.jdk/Contents/Home/bin/java and /Lib
rary/Java/JavaVirtualMachines/jdk1.8.0_101.jdk/Contents/Home/jre/lib/libinstr
ument.dylib. One of the two will be used. Which one is undefined.
[node3 ERROR] objc[1276]: Class JavaLaunchHelper is implemented in both /Libr
ary/Java/JavaVirtualMachines/jdk1.8.0_101.jdk/Contents/Home/bin/java and /Lib
rary/Java/JavaVirtualMachines/jdk1.8.0_101.jdk/Contents/Home/jre/lib/libinstr
ument.dylib. One of the two will be used. Which one is undefined.
[node2 ERROR] objc[1275]: Class JavaLaunchHelper is implemented in both /Libr
ary/Java/JavaVirtualMachines/jdk1.8.0_101.jdk/Contents/Home/bin/java and /Lib
rary/Java/JavaVirtualMachines/jdk1.8.0_101.jdk/Contents/Home/jre/lib/libinstr
ument.dylib. One of the two will be used. Which one is undefined.
ML-ITS-603436:~ dwoodbridge$ ccm status
Cluster: 'test'
---------------
node1: UP
node3: UP
node2: UP
```

Cassandra

Check a node status in a cluster.

- `ccm node_name show`

```
ML-ITS-603436:~ dwoodbridge$ ccm node1 show
node1: UP
       cluster=test
       auto_bootstrap=False
       thrift=('127.0.0.1', 9160)
       binary=('127.0.0.1', 9042)
       storage=('127.0.0.1', 7000)
       jmx_port=7100
       remote_debug_port=0
       byteman_port=0
       initial_token=None
       pid=2051
```

Cassandra

Check a node status in a cluster.

- `ccm node_name show`

```
ML-ITS-603436:~ dwoodbridge$ ccm node2 show
node2: UP
       cluster=test
       auto_bootstrap=False
       thrift=('127.0.0.2', 9160)
       binary=('127.0.0.2', 9042)
       storage=('127.0.0.2', 7000)
       jmx_port=7200
       remote_debug_port=0
       byteman_port=0
       initial_token=None
       pid=2053
```

Cassandra

Check a node status in a cluster.

- `ccm node_name show`

```
ML-ITS-603436:~ dwoodbridge$ ccm node3 show
node3: UP
       cluster=test
       auto_bootstrap=False
       thrift=('127.0.0.3', 9160)
       binary=('127.0.0.3', 9042)
       storage=('127.0.0.3', 7000)
       jmx_port=7300
       remote_debug_port=0
```

Cassandra

Use cqlsh on nodes in a cluster

- `ccm node_name cqlsh`

```
ML-ITS-603436:~ dwoodbridge$ ccm node1 cqlsh
Connected to test at 127.0.0.1:9042.
[cqlsh 5.0.1 | Cassandra 3.9 | CQL spec 3.4.2 | Native protocol v4]
Use HELP for help.
```

- `CREATE KEYSPACE test WITH REPLICATION = {'class': 'SimpleStrategy' , 'replication_factor':2}`
- `USE test;`
- `CREATE TABLE friend(name text, PRIMARY KEY(name));`

Cassandra

Use cqlsh on nodes in a cluster

- `INSERT INTO friend(name) VALUES ('Diane');`
- `INSERT INTO friend(name) VALUES ('Yannet');`
- `INSERT INTO friend(name) VALUES ('David');`
- `INSERT INTO friend(name) VALUES ('Kirsten');`
- `INSERT INTO friend(name) VALUES ('James');`
- `INSERT INTO friend(name) VALUES ('Nathaniel');`
- `INSERT INTO friend(name) VALUES ('Paul');`

```
cqlsh:test> INSERT INTO friend(name) VALUES ('Diane');
cqlsh:test> INSERT INTO friend(name) VALUES ('Yannet');
cqlsh:test> INSERT INTO friend(name) VALUES ('David');
cqlsh:test> INSERT INTO friend(name) VALUES ('Kirsten');
cqlsh:test> INSERT INTO friend(name) VALUES ('James');
cqlsh:test> INSERT INTO friend(name) VALUES ('Nathaniel');
cqlsh:test> INSERT INTO friend(name) VALUES ('Paul');
```

Cassandra

Replication

- Check replicated data.

```
ML-ITS-603436:~ dwoodbridge$ ccm node2 cqlsh
Connected to test at 127.0.0.2:9042.
[cqlsh 5.0.1 | Cassandra 3.9 | CQL spec 3.4.2 | Native protocol v4]
Use HELP for help.
cqlsh> DESCRIBE KEYSPACES;

system_schema  system   system_distributed  system_traces
system_auth    simplex  test                mydb2

cqlsh> use test;
cqlsh:test> SELECT * FROM friend;

 name
-----------
     David
  Nathaniel
     James
    Kirsten
     Diane
      Paul
    Yannet
```

Example

Try to see whether data is replicated to node3

Cassandra

Partitioning (Sharding)

- Check token ring.
  - `ccm node_name ring`

```
ML-ITS-603436:~ dwoodbridge$ ccm node1 ring
objc[9253]: Class JavaLaunchHelper is implemented in both /Library/Java/JavaVirtualMachines/jdk1.8.0_101.jdk/Con
.0_101.jdk/Contents/Home/jre/lib/libinstrument.dylib. One of the two will be used. Which one is undefined.


Datacenter: datacenter1
==========
Address    Rack        Status State   Load            Owns                Token
                                                                          9210101146673835839
127.0.0.1  rack1       Up     Normal  219.76 KiB      ?                   -9204843100512146787
127.0.0.1  rack1       Up     Normal  219.76 KiB      ?                   -9163006488236030522
127.0.0.3  rack1       Up     Normal  218.23 KiB      ?                   -9154855605745133015
127.0.0.3  rack1       Up     Normal  218.23 KiB      ?                   -9144835023218902723
127.0.0.1  rack1       Up     Normal  219.76 KiB      ?                   -9113835427033042364
127.0.0.3  rack1       Up     Normal  218.23 KiB      ?                   -9111941555688496710
127.0.0.3  rack1       Up     Normal  218.23 KiB      ?                   -9105293811387401185
127.0.0.3  rack1       Up     Normal  218.23 KiB      ?                   -9096181527453766903
127.0.0.2  rack1       Up     Normal  214.34 KiB      ?                   -9045984403360323436
127.0.0.2  rack1       Up     Normal  214.34 KiB      ?                   -9043424267273964528
```

Cassandra

Partitioning (Sharding)

- Check token values.

```
cqlsh:test> select name, token(name) from friend;

 name      | system.token(name)
-----------+---------------------
     David | -9107100675643232384
 Nathaniel | -7471314492519062385
     James | -7437409357263336768
   Kirsten | -5252184100220696940
    Diane  | -4488636531397337629
      Paul | -3500811974
    Yannet | -2796904366

(7 rows)
```

```
127.0.0.2  rack1  Up  Normal  214.34 KiB  ?  -4552349328135717413
127.0.0.3  rack1  Up  Normal  218.23 KiB  ?  -4514249937042389638
127.0.0.3  rack1  Up  Normal  218.23 KiB  ?  -4496437166225383212
127.0.0.3  rack1  Up  Normal  218.23 KiB  ?  -4494322534419123084
127.0.0.1  rack1  Up  Normal  219.76 KiB  ?  -4449667641886025618
127.0.0.3  rack1  Up  Normal  218.23 KiB  ?  -4437112008117519062
127.0.0.1  rack1  Up  Normal  219.76 KiB  ?  -4402567310942384003
```

Cassandra

Partitioning (Sharding)

- Check token values.

```
cqlsh:test> select name, token(name) from friend;

 name      | system.token(name)
-----------+---------------------
     David | -9107100675643232384
 Nathaniel | -7471314492519062385
     James | -7437409357263336768
   Kirsten | -5252184100220696940
     Diane | -4488636531397337629
      Paul | -3500811974150555078
    Yannet | -2796904366845320165

(7 rows)
```

```
127.0.0.3  rack1  Up  Normal  218.23 KiB  ?  -2942144691458613302
127.0.0.3  rack1  Up  Normal  218.23 KiB  ?  -2925781883377735539
127.0.0.2  rack1  Up  Normal  214.34 KiB  ?  -2888517123738285216
127.0.0.1  rack1  Up  Normal  219.76 KiB  ?  -2786537904358040890
127.0.0.2  rack1  Up  Normal  214.34 KiB  ?  -2753402426759143918
127.0.0.3  rack1  Up  Normal  218.23 KiB  ?  -2745160886670639336
127.0.0.3  rack1  Up  Normal  218.23 KiB  ?  -2660297764281609632
```

Cassandra

Partitioning (Sharding)

- Check token values.

```
cqlsh:test> select name, token(name) from friend;

 name      | system.token(name)
-----------+---------------------
     David | -9107100675643232384
  Nathaniel | -7471314492519062385
     James | -7437409357263336768
    Kirsten | -5252184100220696940
     Diane | -4488636531397337629
      Paul | -3500811974150555078
    Yannet | -2796904366845320165

(7 rows)
```

```
127.0.0.1  rack1  Up  Normal  219.76 KiB  ?  -9113835427033042364
127.0.0.3  rack1  Up  Normal  218.23 KiB  ?  -9111941555688496710
127.0.0.3  rack1  Up  Normal  218.23 KiB  ?  -9105293811387401185
127.0.0.3  rack1  Up  Normal  218.23 KiB  ?  -9096181527453766903
127.0.0.2  rack1  Up  Normal  214.34 KiB  ?  -9045984403360323436
127.0.0.2  rack1  Up  Normal  214.34 KiB  ?  -9043424267273964528
```

Cassandra

Replication and Sharding - Check an endpoint where data exists for the token.

`ccm node_name nodetool getendpoints keyspace_name table_name key_value`

```
ML-ITS-603436:~ dwoodbridge$ ccm node1 nodetool getendpoints test friend Diane
objc[6768]: Class JavaLaunchHelper is implemented in both /Library/Java/JavaVirtualMachines/jdk1.8.0_101.j
dk/Contents/Home/bin/java and /Library/Java/JavaVirtualMachines/jdk1.8.0_101.jdk/Contents/Home/jre/lib/lib
instrument.dylib. One of the two will be used. Which one is undefined.

127.0.0.1
127.0.0.3
```

Sharded and Replicated!

```
ML-ITS-603436:~ dwoodbridge$ ccm node1 nodetool getendpoints test friend Yannet
objc[6799]: Class JavaLaunchHelper is implemented in both /Library/Java/JavaVirtualMachines/jdk1.8.0_101.j
dk/Contents/Home/bin/java and /Library/Java/JavaVirtualMachines/jdk1.8.0_101.jdk/Contents/Home/jre/lib/lib
instrument.dylib. One of the two will be used. Which one is undefined.

127.0.0.1
127.0.0.2
```

Sharded and Replicated!

```
ML-ITS-603436:~ dwoodbridge$ ccm node1 nodetool getendpoints test friend David
objc[6830]: Class JavaLaunchHelper is implemented in both /Library/Java/JavaVirtualMachines/jdk1.8.0_101.j
dk/Contents/Home/bin/java and /Library/Java/JavaVirtualMachines/jdk1.8.0_101.jdk/Contents/Home/jre/lib/lib
instrument.dylib. One of the two will be used. Which one is undefined.

127.0.0.3
127.0.0.2
```

Sharded and Replicated!

http://docs.datastax.com/en/cassandra/2.0/cassandra/tools/toolsGetEndPoints.html

Example

Can you read data from the nodes that does not serve as a replica?

- Can you query “Diane” on node2?

Cassandra

Tunable Consistency

- Setting `Consistency` will ensure the majority of the nodes to respond to read/write.
  - Read: The column with the newest timestamp is returned back to the client.
  - Write: The new update will be propagate to the majority of the nodes.
  - Available consistency levels : ALL, EACH_QUORUM, QUORUM, LOCAL_QUORUM, ONE, TWO, THREE, LOCAL_ONE, ANY, SERIAL, LOCAL_SERIAL.

```
cqlsh> consistency;
Current consistency level is ONE.
cqlsh> consistency QUORUM;
Consistency level set to QUORUM.
```

ONE is default.

https://docs.datastax.com/en/cassandra/2.0/cassandra/dml/dml_config_consistency_c.html

Cassandra Interview Questions

Cassandra data model

Tunable consistency (Quorum)

Difference between RDBMS and Cassandra

CAP Theorem

Cassandra query language

cqlsh

Compaction

Super column

Column family

Memtable

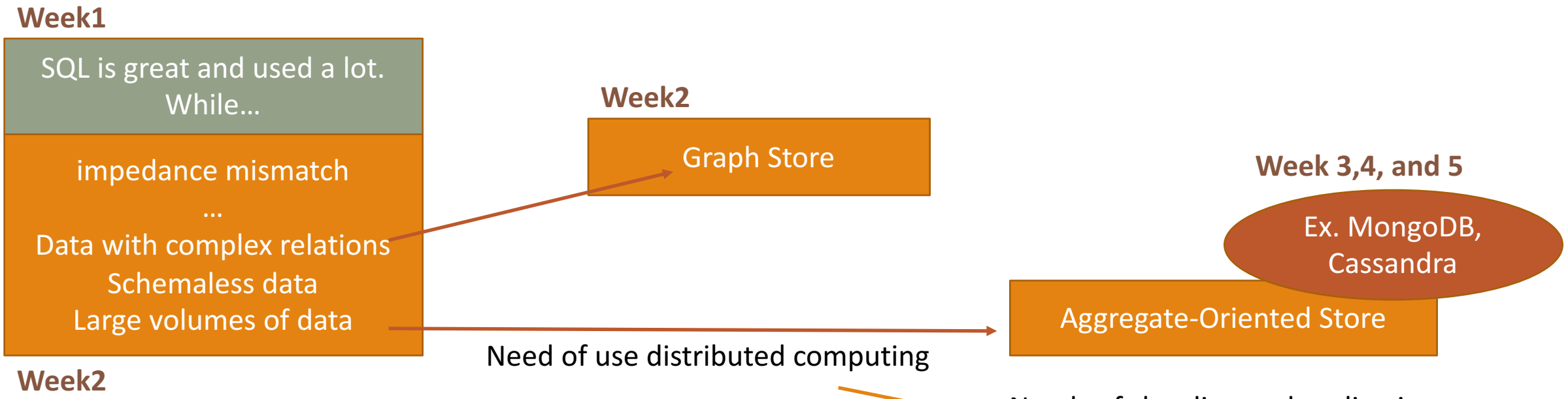
SSTable

Keyspace

Tombstone

Summary

Remember the big picture!



Summary

NoSQL

Pros (In general)
Mostly open-source.
Schemaless.
Good for non-relational data.
Scalable.
Runs well on distributed systems.

Cons
Installation, toolsets still maturing.

Example
Redis, MongoDB, Cassandra, OrientDB

Two main reasons to consider NoSQL.

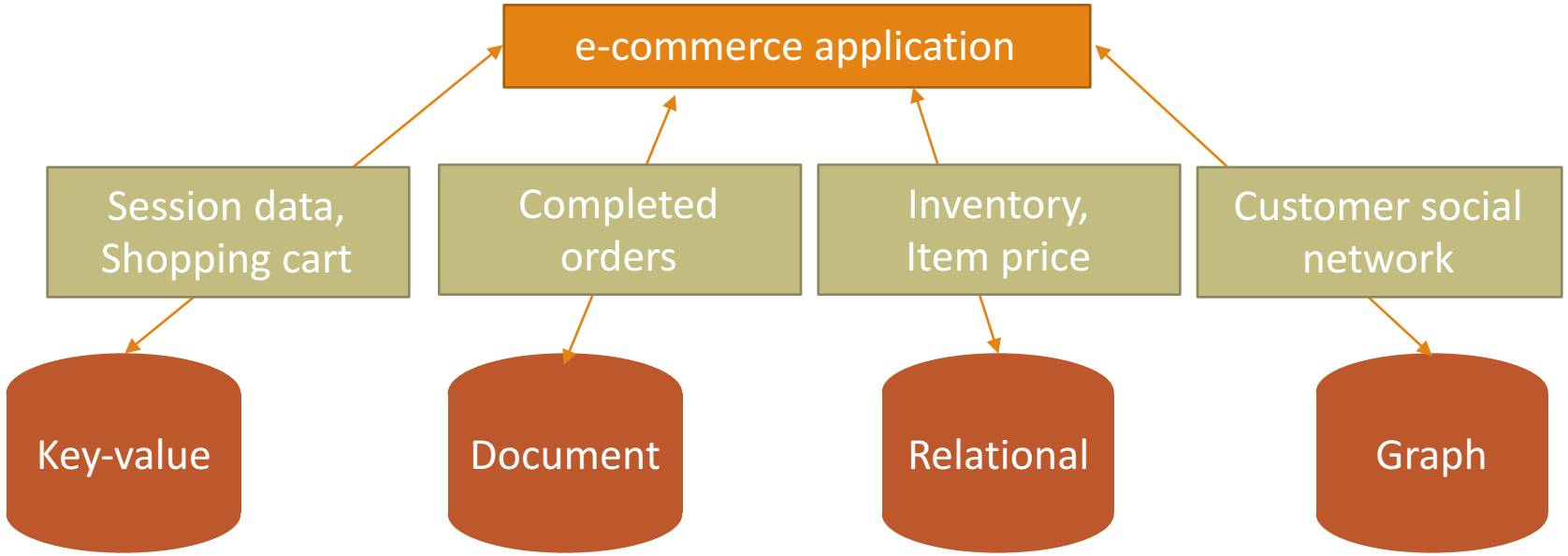
- To improve programmer productivity that better matches an application's needs.
- To improve data access performance via some combination of handling larger data volumes, reducing latency and improving throughput.

Test your expectations about programmer productivity and performance before committing to using a NoSQL technology.

Summary

Polygot Persistence

- Using multiple data storage technologies, chosen based on the way data is being used by individual applications.
- NoSQL data stores do not replace relational databases.



Final Exam

Time : Dec 9th, 10 AM - 12 PM, 101 Howard 154, 155 and 156.

Topic : Week 1 – Week 7.

Type of Questions : Focusing on

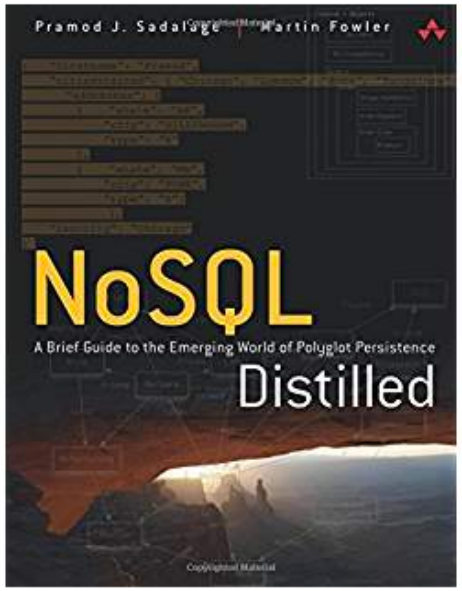
- the concepts of overall NoSQL technologies (Similar to Quiz 1-3, Multiple Choices similar to Paul's Final)
- Programming questions.
  - Ex. What is a problem of the following code (2pt) and why (2pt) and how to fix it (2pt)?

```
cqlsh:test> CREATE TABLE friend(name text);
```

Further Readings

As we discussed in our 1st week lecture, I'd recommend NoSQL Distilled (Not required, but will be helpful for your interviews) and online documentation for Mongo and Cassandra as they are still evolving.

- All the references have been cited in the footnotes or reference section in every slide!



Relational Database Interview Questions

- Basic Operations
  - Create
  - Insert
  - Select
  - Update
  - Join (Inner, Outer, Left, Right, etc.)*
  - Union All/Union/ Union Distinct
  - Minus
  - Intersect
- Normalization (1NF, 2NF and 3NF)*
- Transaction(Concurrency Control) – ACID*
- Indexing* - B tree, Hash
- Truncate vs Delete
- Difference Where vs. Having

NoSQL Interview Questions

What is NoSQL?

Relational Database vs. NoSQL

Impedence mismatch

Polygot persistence

Aggregate-oriented database

Key-value database

Document database

Column family database

Graph database

Replication vs sharding

CAP Theorem

Eventual Consistency

Map-Reduce

MongoDB Interview Questions

MongoDB's type

MongoDB's characteristics

Alternative databases

Supported programming languages

Index

Aggregation Operations(aggregation pipeline)

Sharding

Replication

GridFS

ObjectId

Consistency

Cassandra Interview Questions

Cassandra data model

Tunable consistency (Quorum)

Difference between RDBMS and Cassandra

CAP Theorem

Cassandra query language

cqlsh

Compaction

Super column

Column family

Memtable

SSTable

Keyspace

Tombstone

References

Sadalage, Pramod J., and Martin Fowler. *NoSQL distilled: a brief guide to the emerging world of polyglot persistence*. Pearson Education, 2012.

Apache Cassandra Documentation v4.0, http://cassandra.apache.org/doc/latest/, 2016.

Apache Cassandra Documentation, https://docs.datastax.com/en/cassandra/, 2016

Carpenter, Jeff, and Eben Hewitt. *Cassandra: The Definitive Guide*. " O'Reilly Media, Inc.", 2016.