SNS COLLEGE OF TECHNOLOGY
( An Autonomous Institution)
COIMBATORE–35


DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

# BIG DATA ANALYTICS

NIMMY PRABHA
AP/AIML

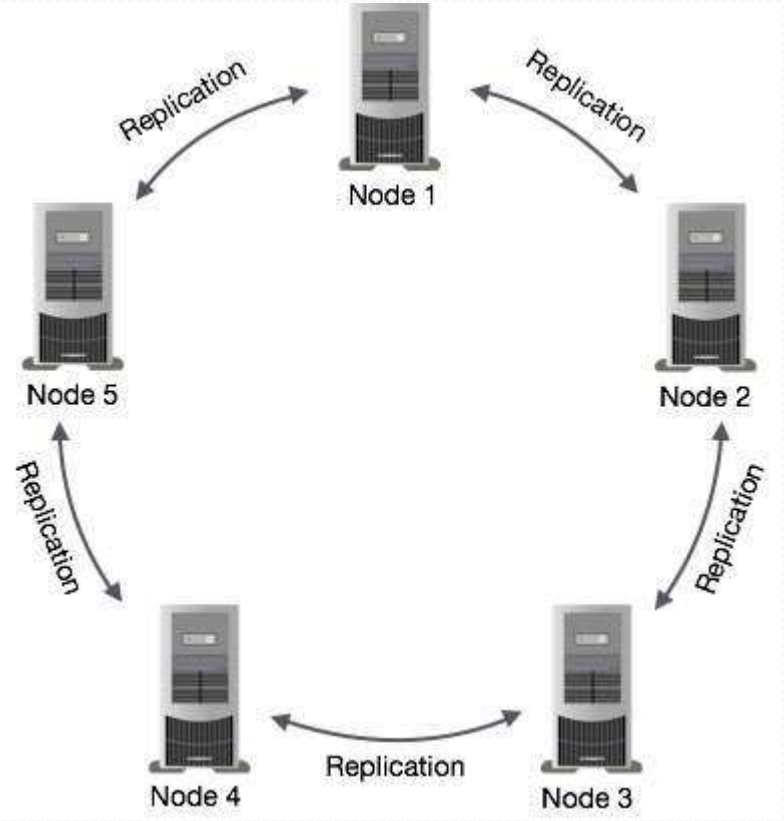
# Features of Cassandra

NIMMY PRABHA

# Features

- Peer-to-peer Network
- Gossip and Failure Detection
- Practitioner
- Replication factor
- Anti-Entropy and Read Repair
- Writes
- Hinted Handoffs
- Tunable consistency - Strong ,Eventual , Read, write

# Data replication in Cassandra



# Components of Cassandra

| | |
|---|---|
| **Node** | • It is the place where data is stored. |
| **Data center** | • It is a collection of related nodes. |
| **Cluster** | • A cluster is a component that contains one or more data centers. |
| **Commit log** | • The commit log is a crash-recovery mechanism in cassandra. Every write operation is written to the commit log. |
| **Mem-table** | • A mem-table is a memory-resident data structure. After commit log, the data will be written to the mem-table. Sometimes, for a single-column family, there will be multiple mem-tables. |
| **SSTable** | • It is a disk file to which the data is flushed from the mem-table when its contents reach a threshold value. |
| **Bloom filter** | • These are nothing but quick, nondeterministic, algorithms for testing whether an element is a member of a set. It is a special kind of cache. Bloom filters are accessed after every query |

# Cassandra Query Language

- **Write Operations**
  - Every write activity of nodes is captured by the **commit logs** written in the nodes. Later the data will be captured and stored in the **mem-table**
- **Read Operation**
  - During read operations, Cassandra gets values from the mem-table and checks the bloom filter to find the appropriate SSTable that holds the required data.

# QL Data types

| CQL Type | Constants supported | Description |
|---|---|---|
| ascii | strings | US-ASCII character string |
| bigint | integers | 64-bit signed long |
| blob | blobs | Arbitrary bytes (no validation), expressed as hexadecimal |
| boolean | booleans | true or false |
| counter | integers | Distributed counter value (64-bit long) |
| date | strings | Value is a date with no corresponding time value; Cassandra encodes date as a 32-bit integer representing days since epoch (January 1, 1970). Dates can be represented in queries and inserts as a string, such as 2015-05-03 (yyyy-mm-dd) |
| decimal | integers, floats | Variable-precision decimal<br>Java type<br>**Note:** When dealing with currency, it is a best practice to have a currency class that serializes to and from an int or use the Decimal form. |
| double | integers, floats | 64-bit IEEE-754 floating point<br><br>Java type |
| float | integers, floats | 32-bit IEEE-754 floating point<br><br>Java type |

| | | |
|---|---|---|
| frozen | user-defined types, collections, tuples | A frozen value serializes multiple components into a single value. Non-frozen types allow updates to individual fields. Cassandra treats the value of a frozen type as a blob. The entire value must be overwritten.<br>**Note:** Cassandra no longer requires the use of frozen for tuples:<br>frozen <tuple <**int**, tuple<text, **double**>>> |
| inet | strings | IP address string in IPv4 or IPv6 format, used by the python-cql driver and CQL native protocols |
| int | integers | 32-bit signed integer |
| list | n/a | A collection of one or more ordered elements: [literal, literal, literal]. |
| map | n/a | A JSON-style array of literals: { literal : literal, literal : literal ... } |
| set | n/a | A collection of one or more elements: { literal, literal, literal } |
| smallint | integers | 2 byte integer |
| text | strings | UTF-8 encoded string |
| time | strings | Value is encoded as a 64-bit signed integer representing the number of nanoseconds since midnight. Values can be represented as strings, such as 13:30:54.234. |
| timestamp | integers, strings | Date and time with millisecond precision, encoded as 8 bytes since epoch. Can be represented as a string, such as 2015-05-03 13:30:54.234. |