

DataStax Developer Days



Introduction
to
Cassandra

You've
Worked
Hard





**Finally, it's
Taking off!**



**Are
You
Ready?**

Well,
You
May be
Ready...



But,
What about
Your Data?



Start with “Why?”

Why do I need DSE & Apache Cassandra™?

- Only DataStax Enterprise offers CARDS – and you need it all:
 - **Contextual** – relevant data in context
 - **Available** – always on, no downtime
 - **Realtime** – response time in MS
 - **Distributed** – many servers in datacenters around the world
 - **Scalable** – near linear increase for each additional server

Why?



How does Cassandra do it?

Here's our agenda:

- We'll take a quick look at a Cassandra cluster
- Then, we'll discuss Cassandra's architectural
- Finally, we'll do some more hands-on
 - Load some data
 - Write CQL queries against the data
 - Demonstrate the power of replication



Apache Cassandra™ First Touch

What Does Cassandra Look Like?

The basic structure of data

Cartoon Keyspace

Cartoon Characters Table				Episode Table				RatingsTable			
42	---	---	---	---	---	---	---	---	---	---	---
17	---	---	---	---	---	---	---	---	---	---	---
83	---	---	---	---	---	---	---	---	---	---	---
92	---	---	---	---	---	---	---	---	---	---	---
37	---	---	---	---	---	---	---	---	---	---	---
47	---	---	---	---	---	---	---	---	---	---	---
22	---	---	---	---	---	---	---	---	---	---	---
38	---	---	---	---	---	---	---	---	---	---	---
93	---	---	---	---	---	---	---	---	---	---	---
18	---	---	---	---	---	---	---	---	---	---	---
63	---	---	---	---	---	---	---	---	---	---	---
71	---	---	---	---	---	---	---	---	---	---	---

Keyspaces
Contain
Tables

What Does Cassandra Look Like?

The basic structure of data

Cartoon Characters Table

	Last Name	First Name	Address	Email
42	---	---	---	---
	---	---	---	---
	---	---	---	---
	---	---	---	---
	---	---	---	---
	---	---	---	---
17	---	---	---	---
	---	---	---	---
	---	---	---	---
	---	---	---	---
83	---	---	---	---
	---	---	---	---
	---	---	---	---
	---	---	---	---
92	---	---	---	---
	---	---	---	---
	---	---	---	---

Episode Table

	Season	Episode	Name	Time
37	---	---	---	---
	---	---	---	---
	---	---	---	---
	---	---	---	---
	---	---	---	---
47	---	---	---	---
	---	---	---	---
	---	---	---	---
	---	---	---	---
22	---	---	---	---
	---	---	---	---
	---	---	---	---
38	---	---	---	---
	---	---	---	---
	---	---	---	---

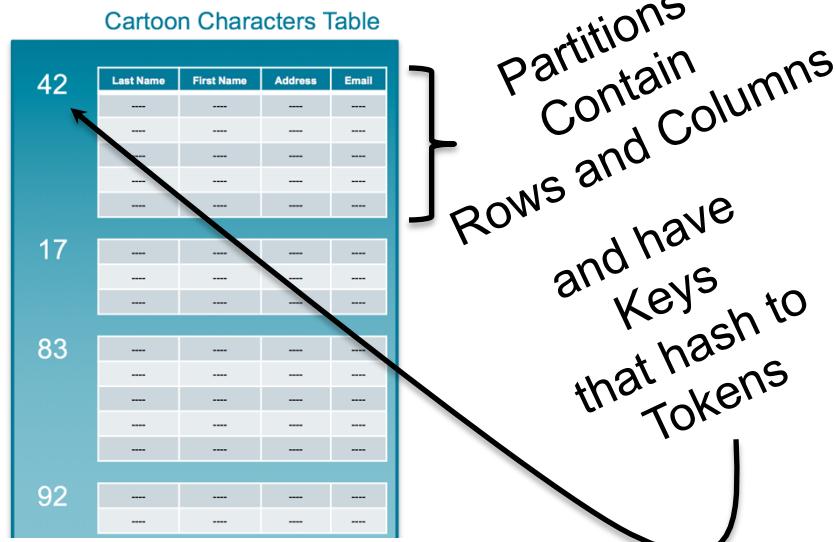
RatingsTable

	Cartoon	Season	Episode	Stars
93	---	---	---	---
	---	---	---	---
	---	---	---	---
	---	---	---	---
18	---	---	---	---
	---	---	---	---
	---	---	---	---
	---	---	---	---
63	---	---	---	---
	---	---	---	---
	---	---	---	---
71	---	---	---	---
	---	---	---	---
	---	---	---	---

Tables
Contain
Partitions

What Does Cassandra Look Like?

The basic structure of data



Want to try it?

Let's look at a cluster

- Open a browser
- Go to <http://<your IP address>:9091>
- Open notebook
 - "Core Cassandra: First Touch"



Apache Cassandra™ First Touch

Quick review

- Key Take-aways:
 - Clusters contain keyspaces
 - Keyspaces contain tables
 - Tables contain partitions
 - Partitions contain rows and columns
 - CQL has syntax similar to SQL

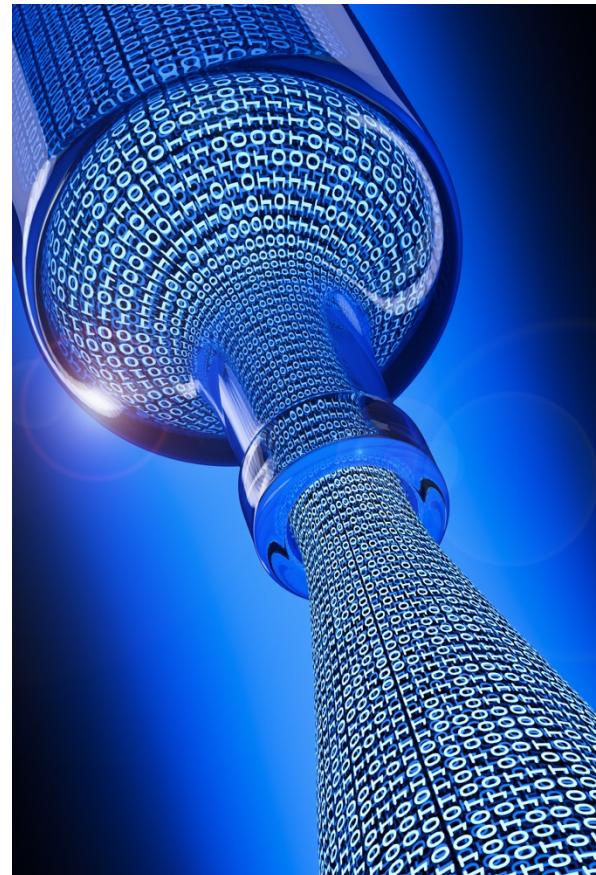
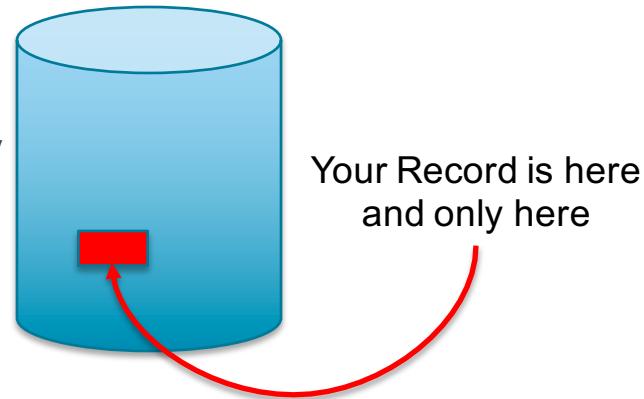


Cassandra Architecture

Problems with Traditional RDBM

Each record is in one location

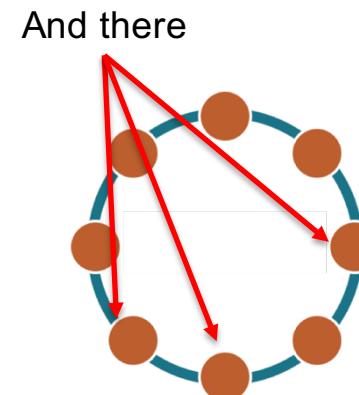
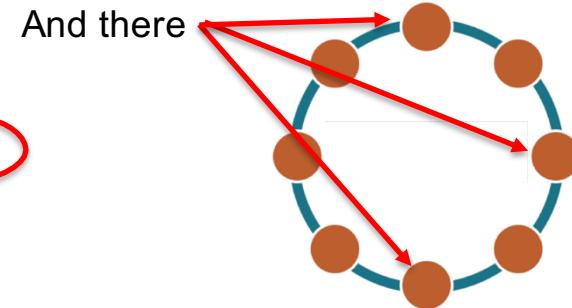
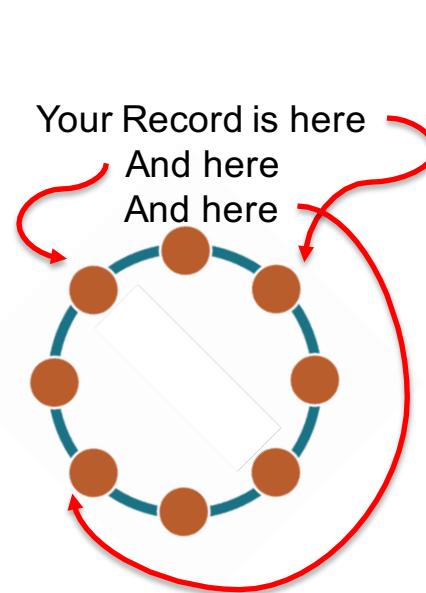
- Update in place – causes bottlenecks
 - Reduced throughput and latency
- Single point of failure
 - Availability risk
- Dogmatic consistency



Cassandra

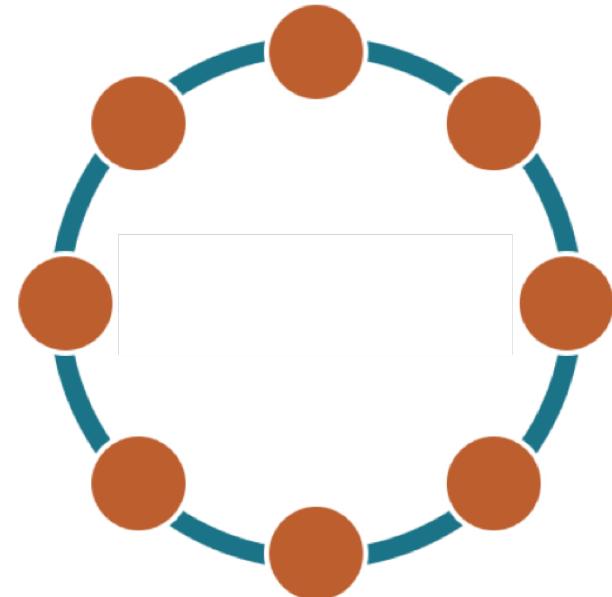
Features

- Distributed
 - Available
 - Responsive
 - Scalable
- Log-structured
 - No bottlenecks
- Tunable consistency



Cassandra is Distributed

- Cassandra clusters have many nodes



Cassandra is Distributed

- Cassandra clusters have many nodes
- How does Cassandra manage all the nodes?



Cassandra is Distributed

- Each cluster has many nodes
- How does Cassandra manage all the nodes?
 - There is no boss-node



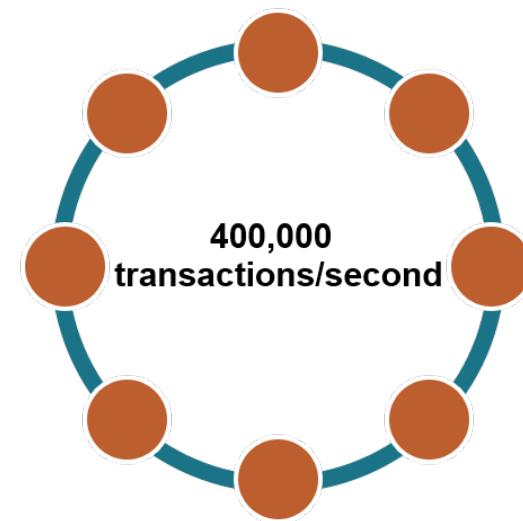
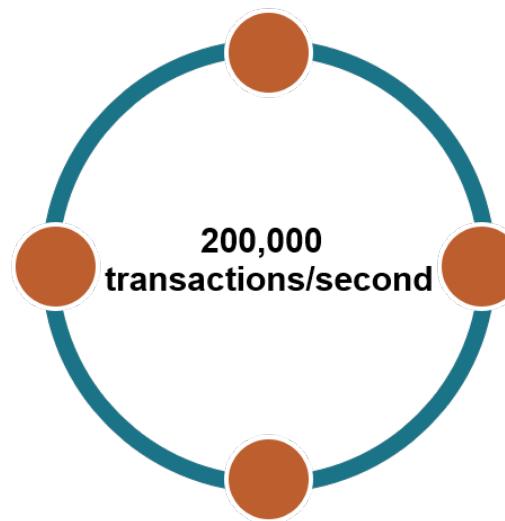
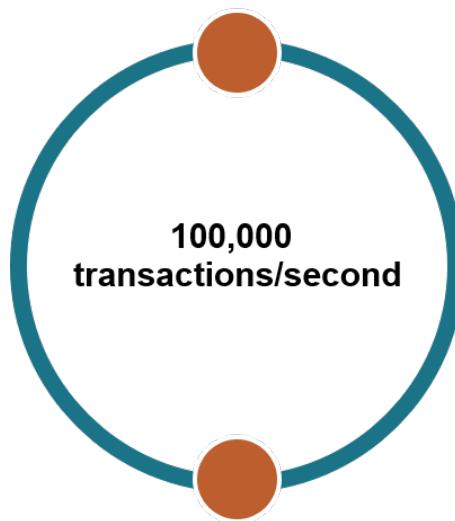
Cassandra is Distributed

- Cassandra clusters have many nodes
- How does Cassandra manage all the nodes?
 - There is no boss-node
 - The nodes collaborate



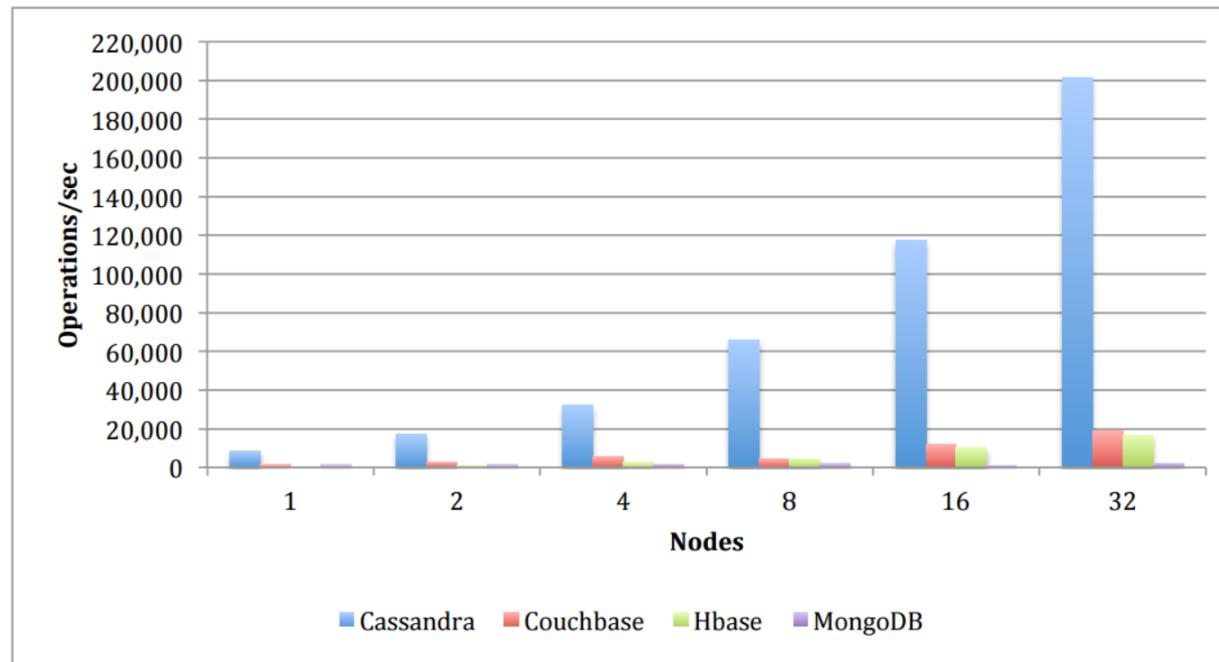
Horizontal vs. Vertical Scaling

- Vertical scaling requires one large expensive machine
- Horizontal scaling requires multiple less-expensive commodity hardware



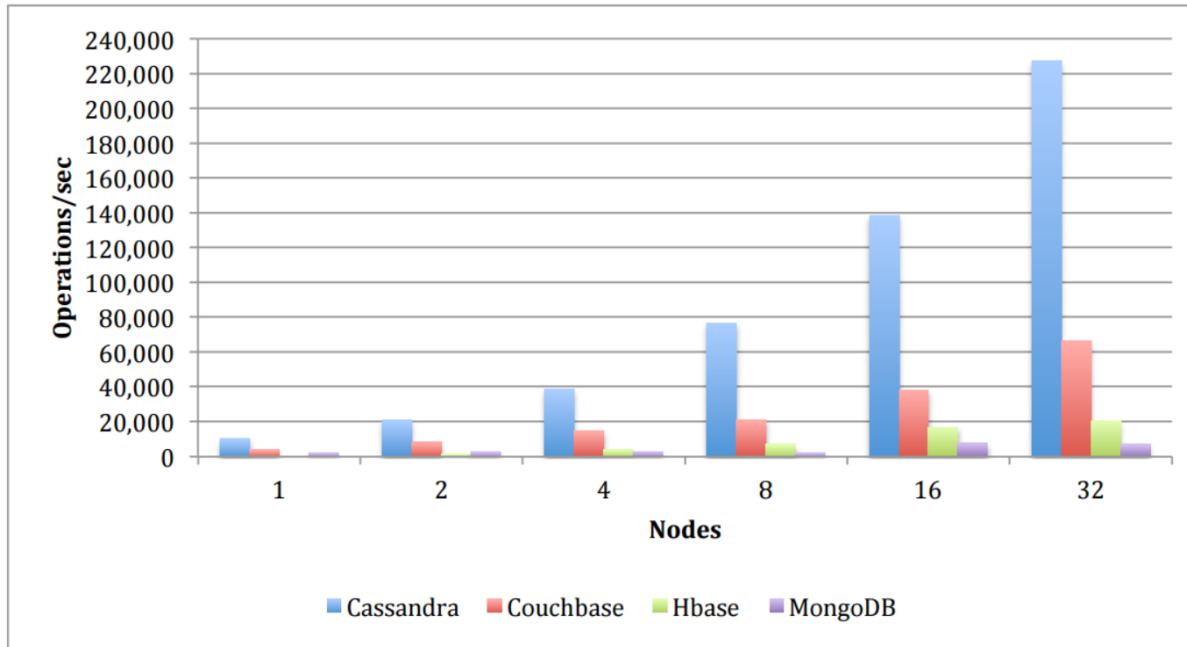
Scales Linearly

Read-Modify-Write Workload



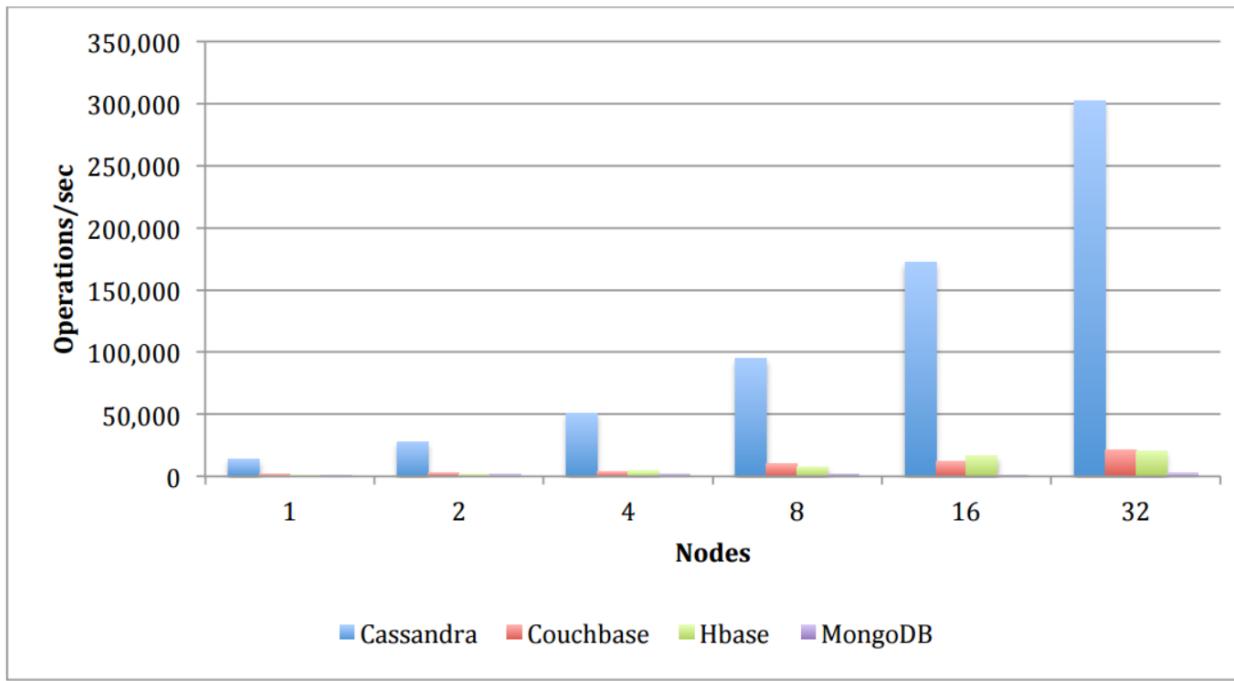
Scales Linearly

Read-mostly Workload



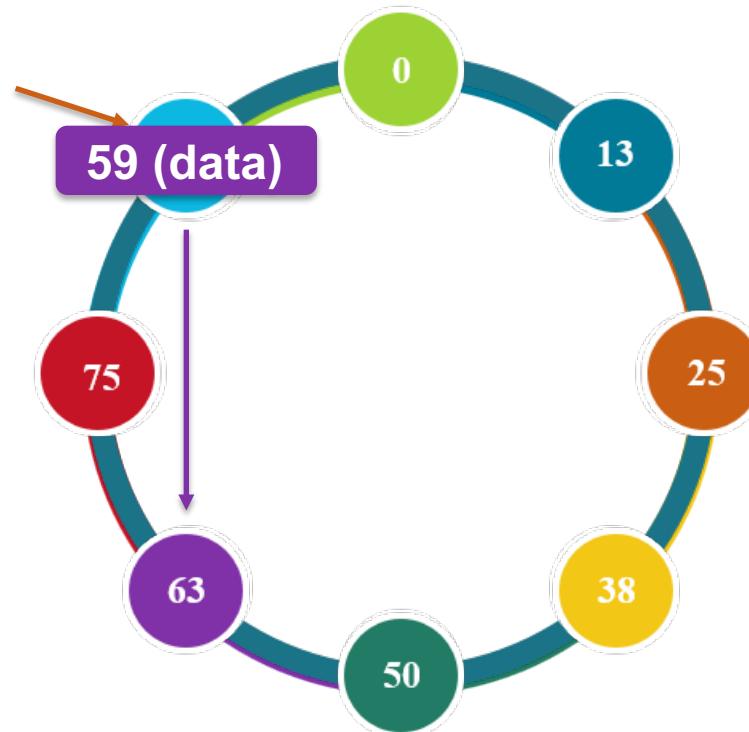
Scales Linearly

Balanced Read/Write Mix

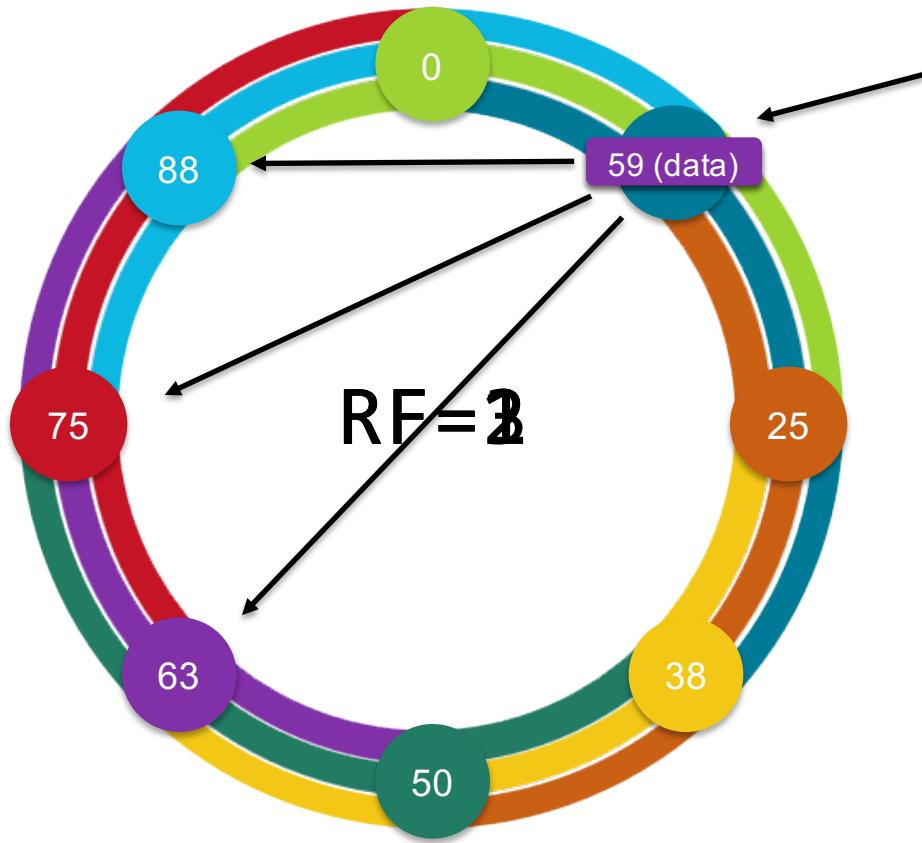


Cassandra's Token Ring and Data Replication

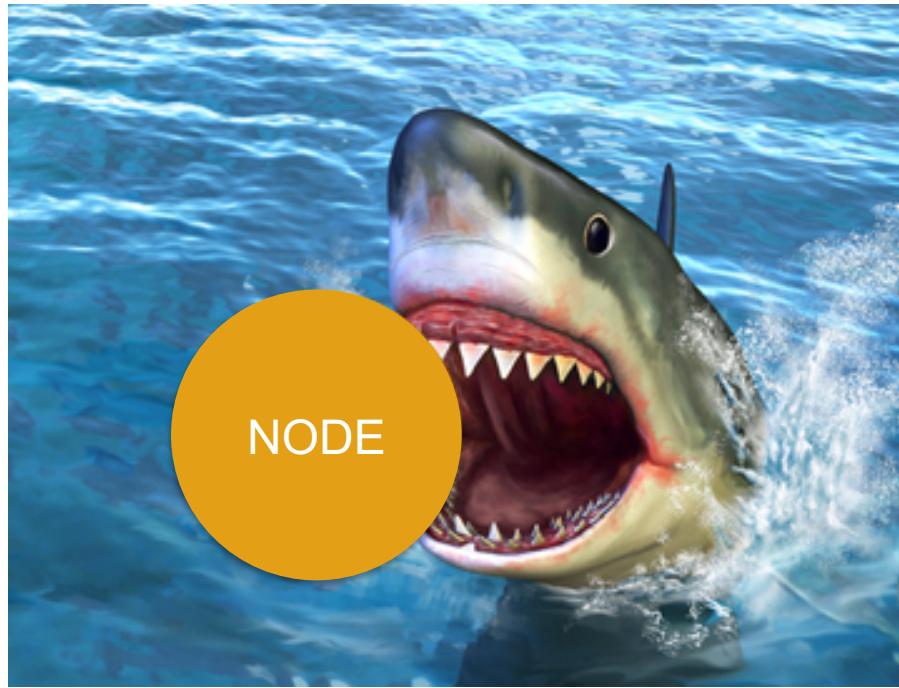
How the Ring Works



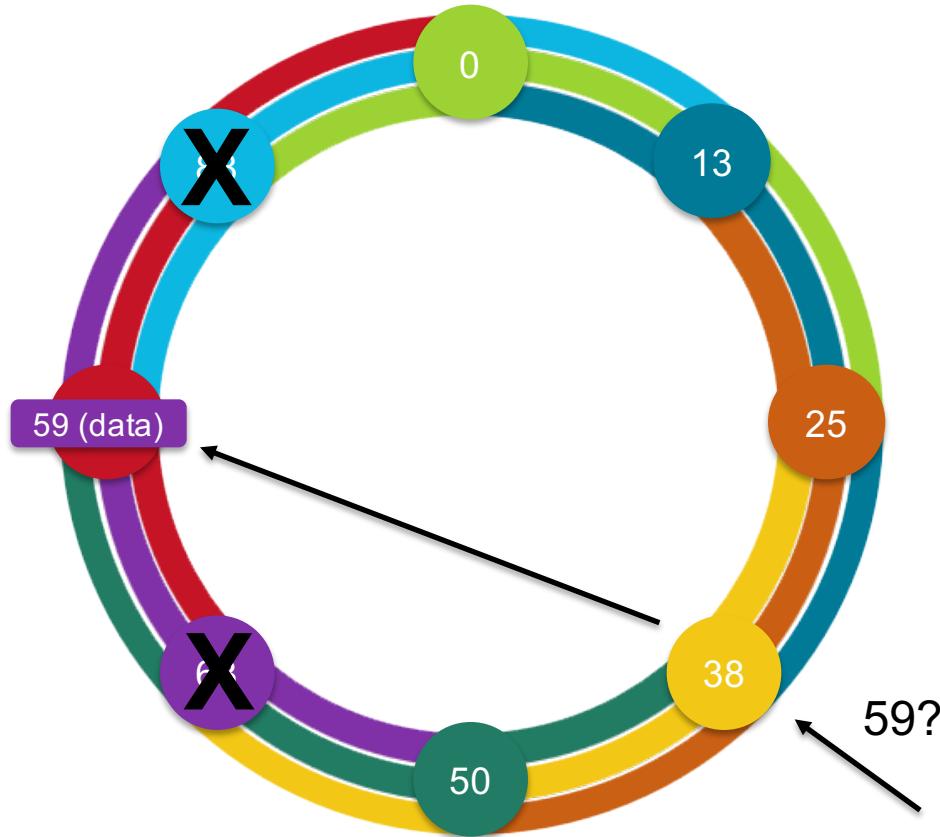
Replication within the Ring



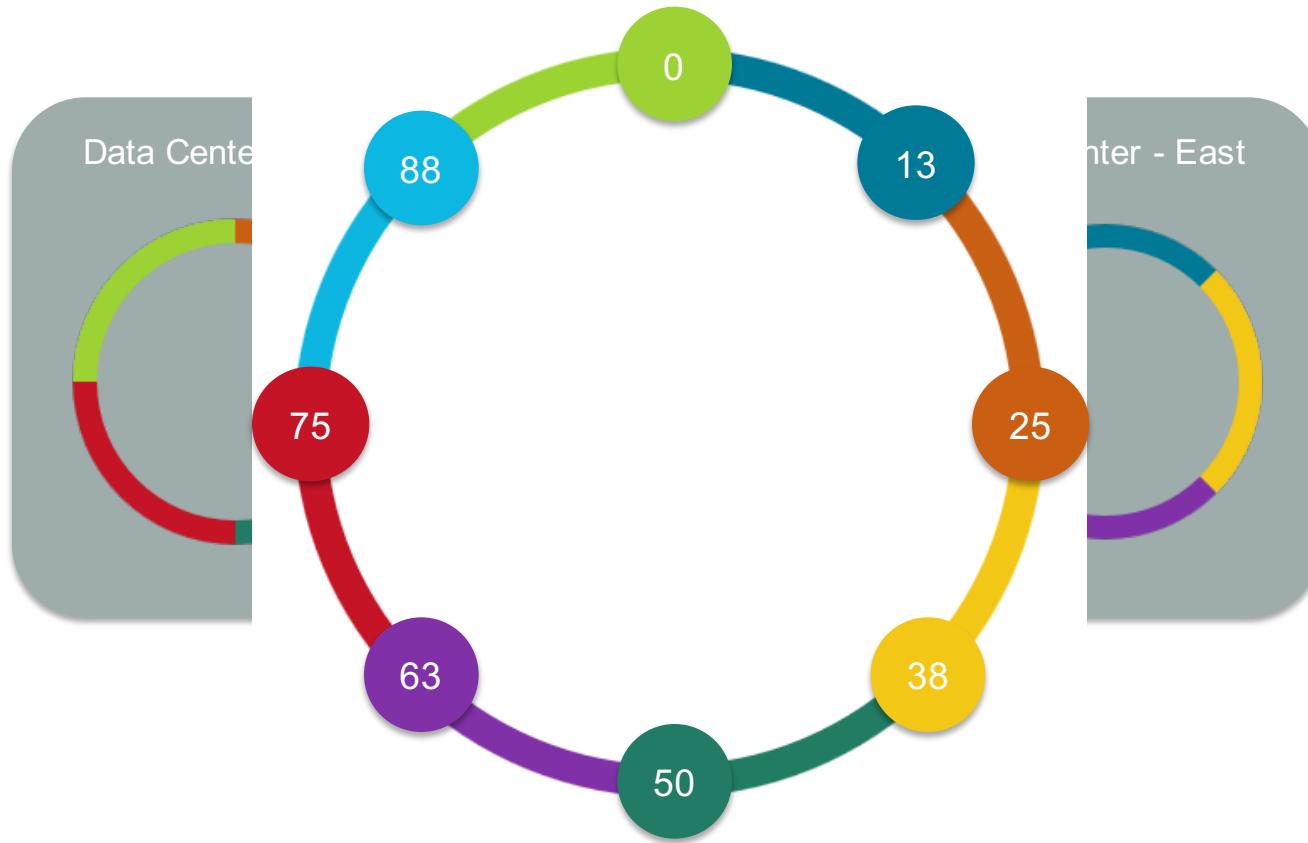
Node Failure



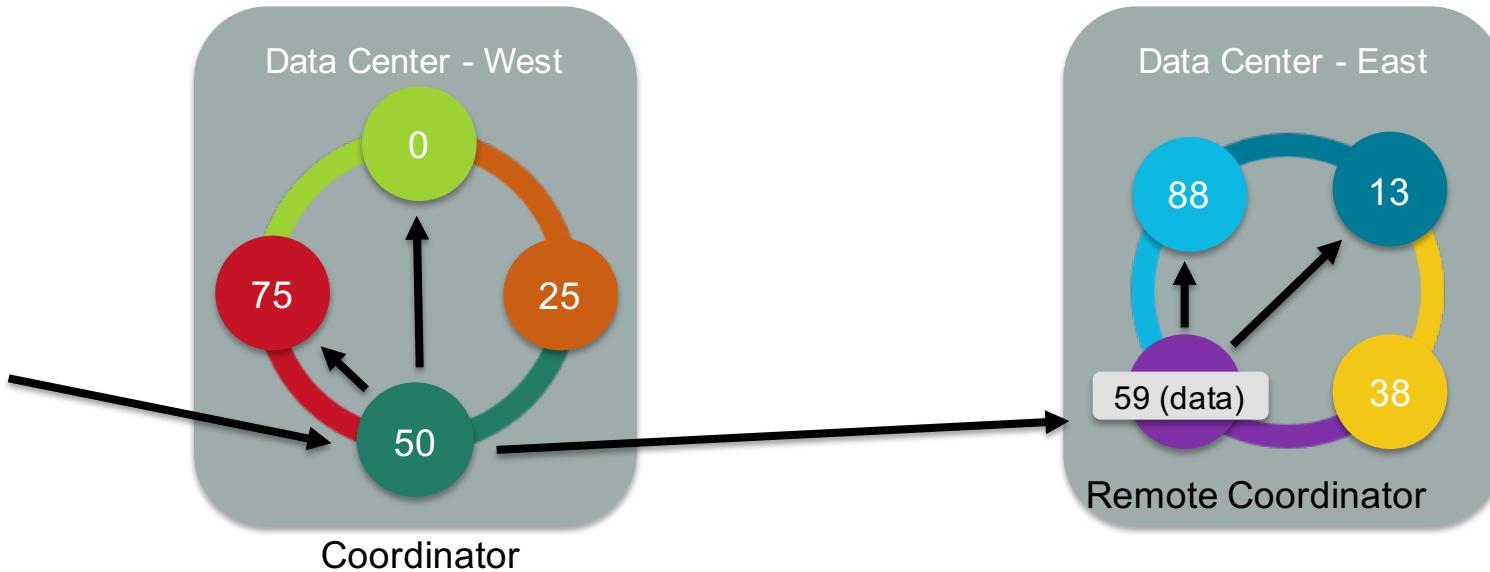
Replication



Multi-Data Center Replication

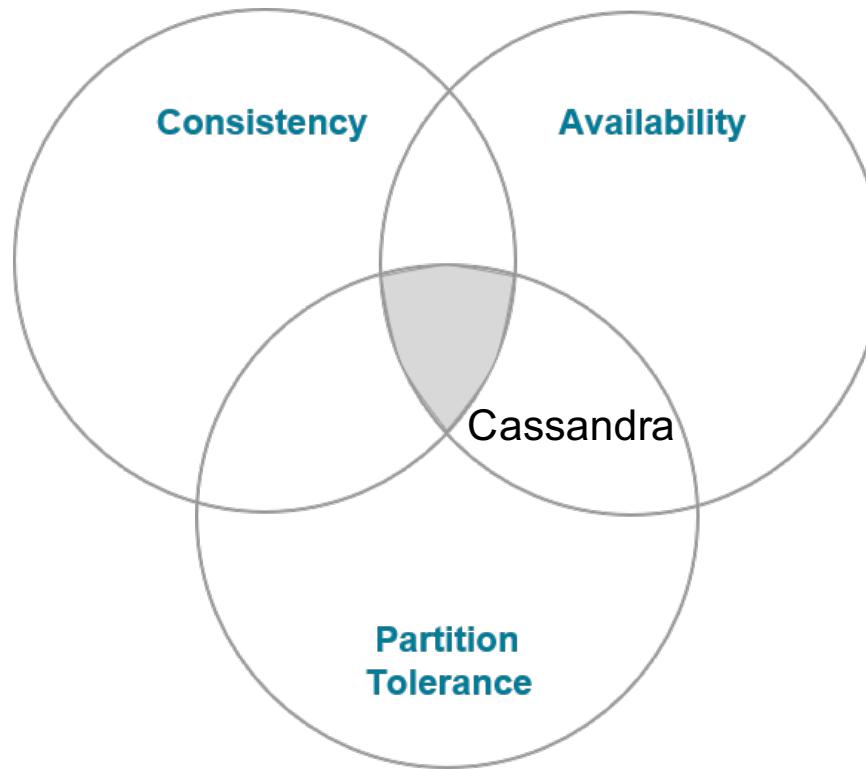


Multi-Data Center Replication

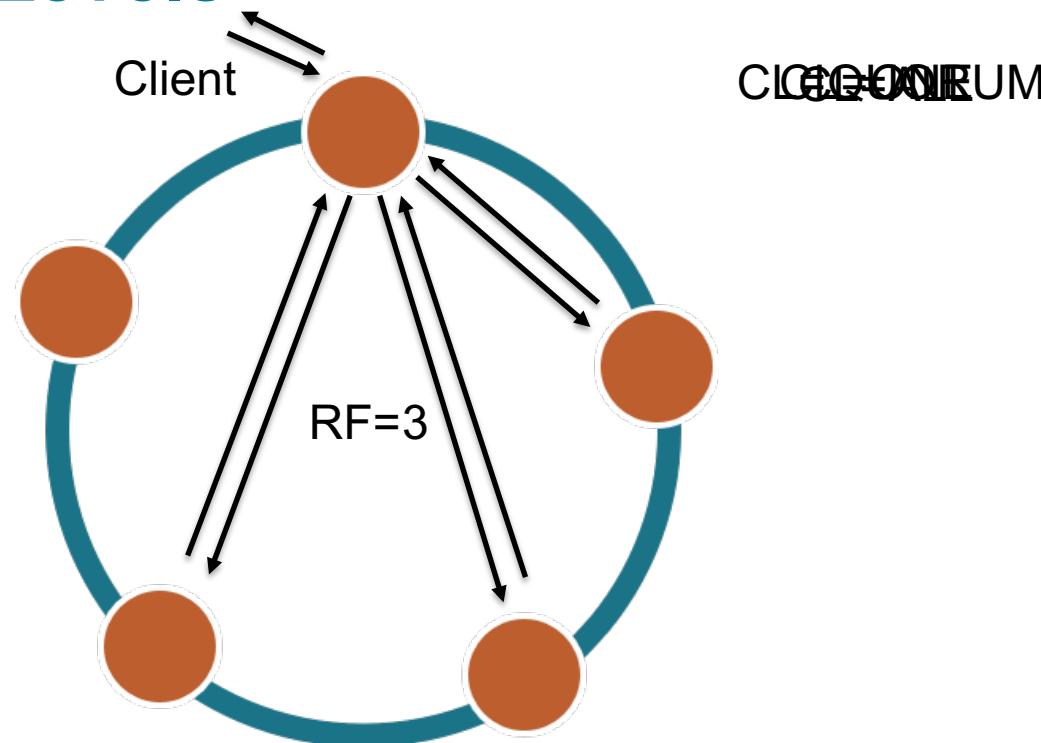


Cassandra's Consistency

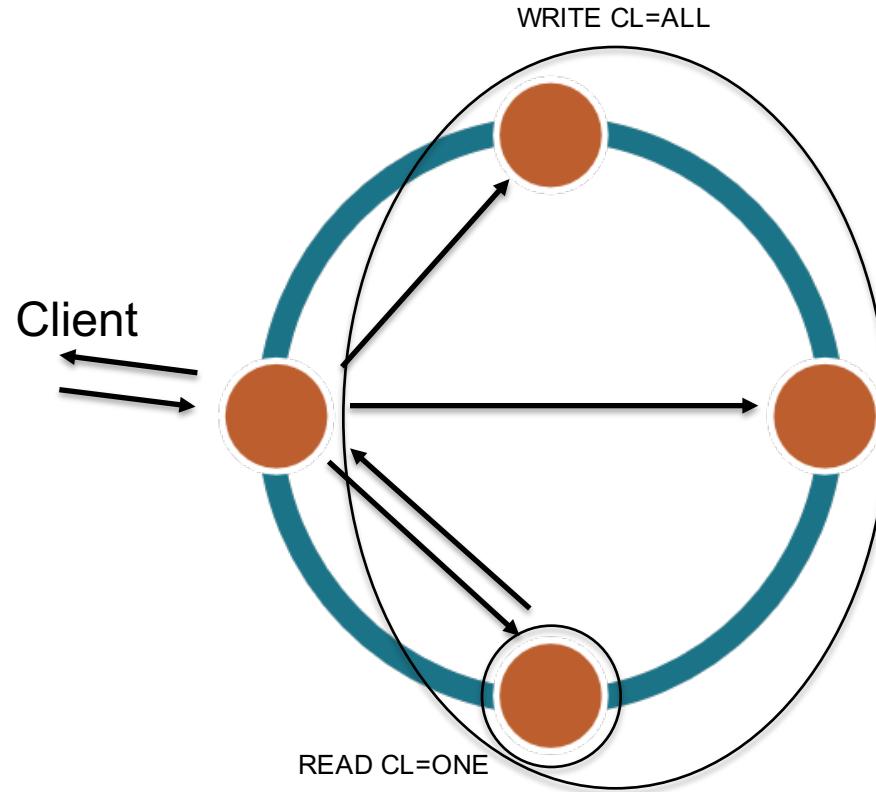
CAP Theorem



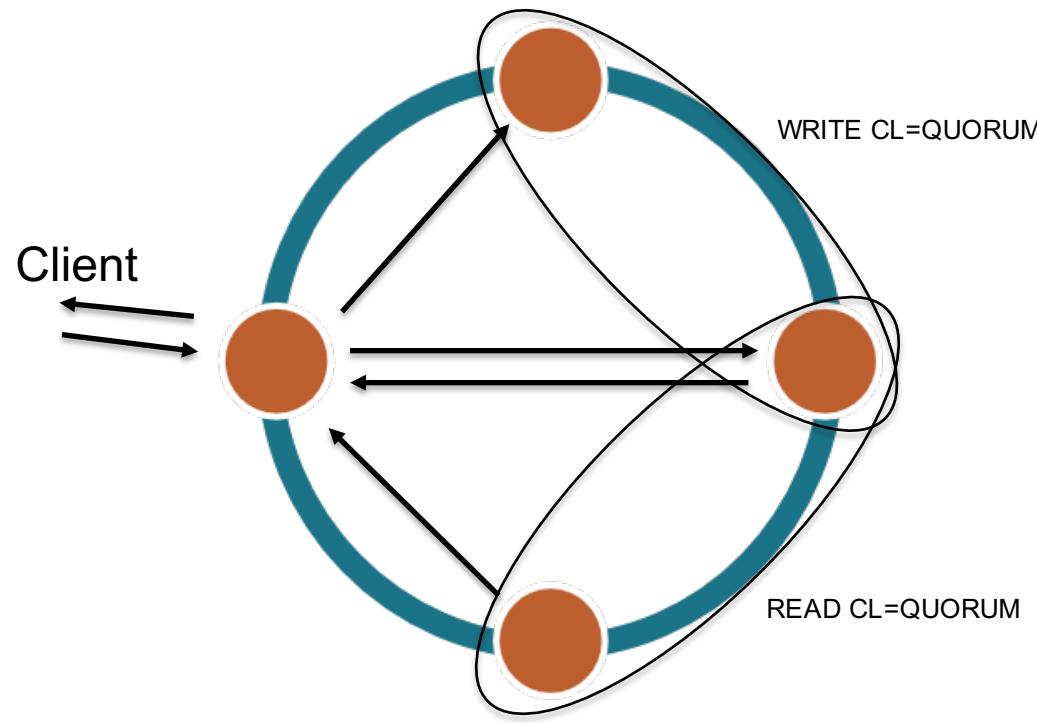
Consistency Levels



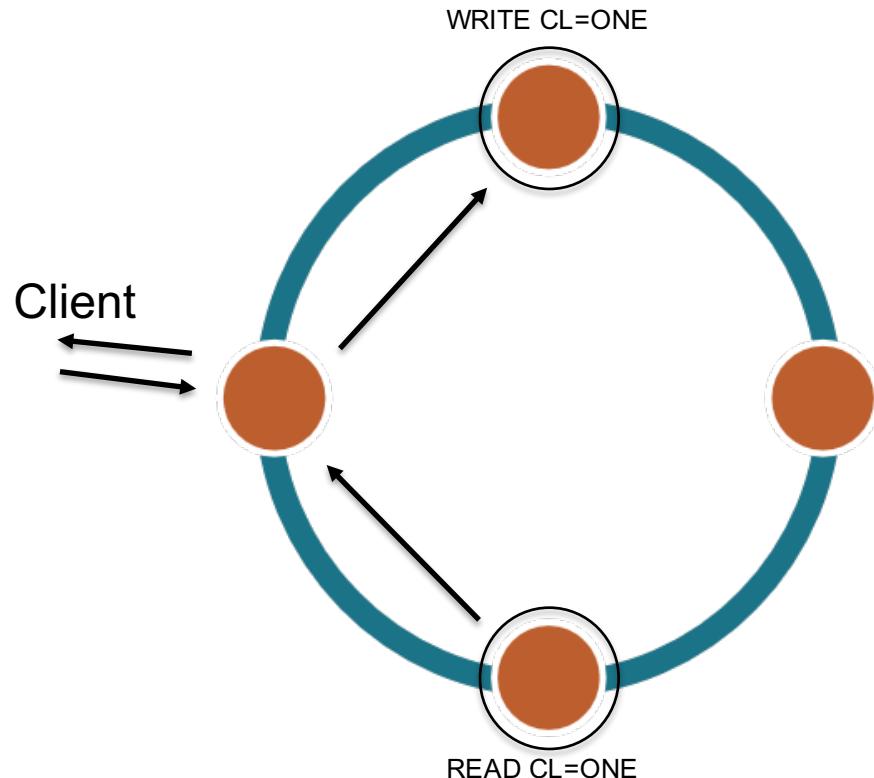
Strong Consistency



Quorum



CL=ONE



Log-Structure

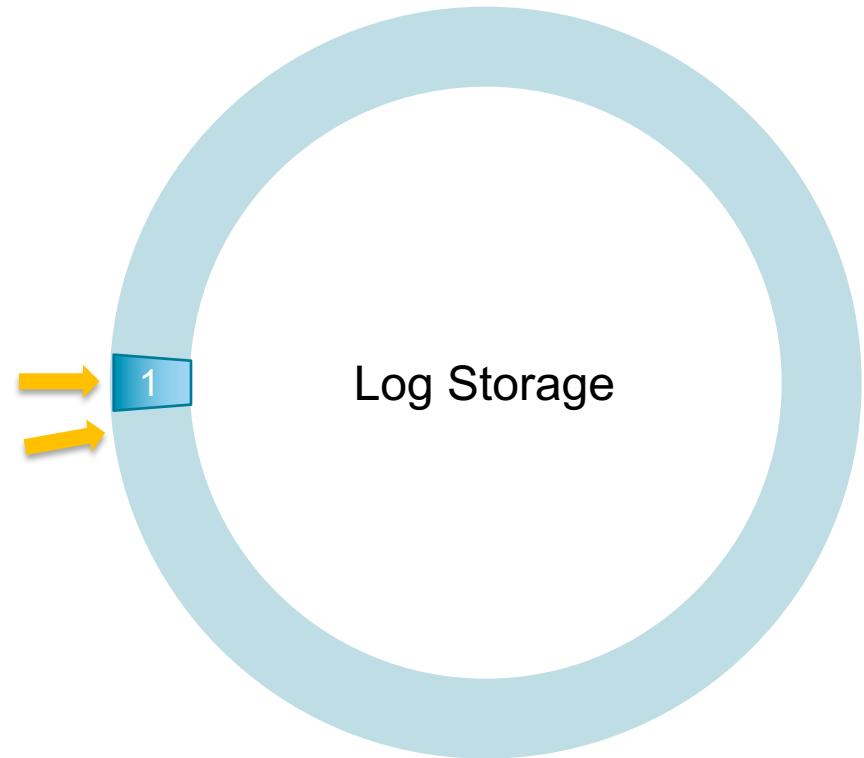
What is Log-Structured?

Not This!



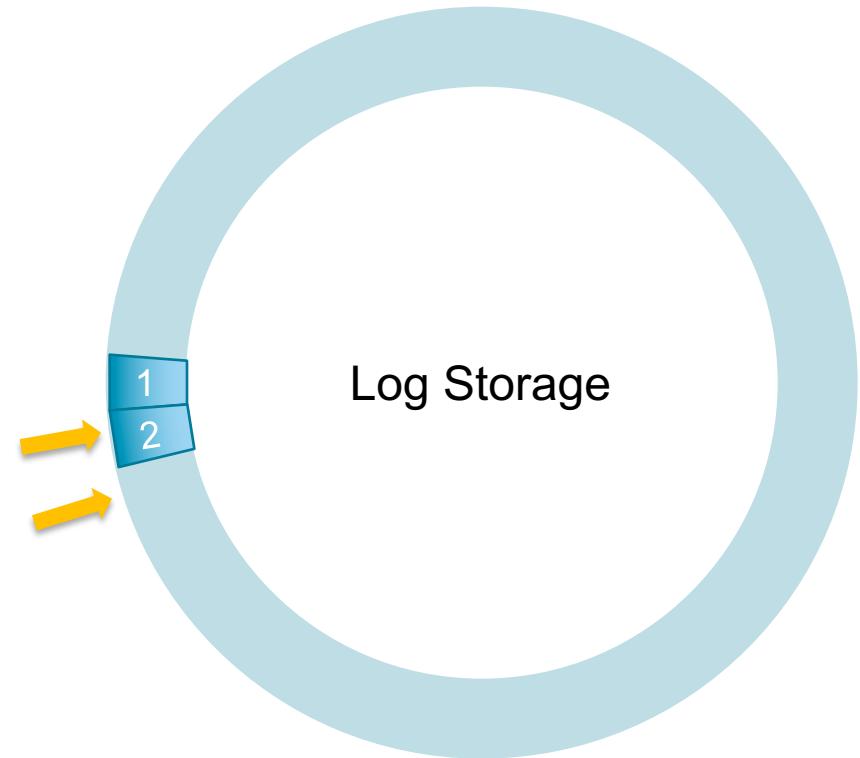
What is Log-Structured?

Record 1



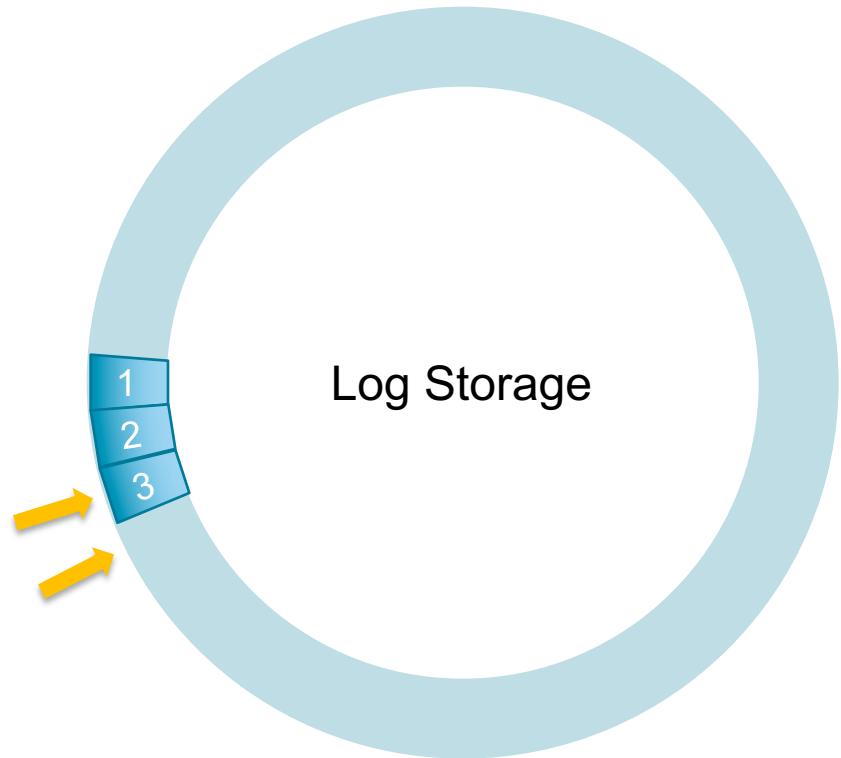
What is Log-Structured?

Record 2



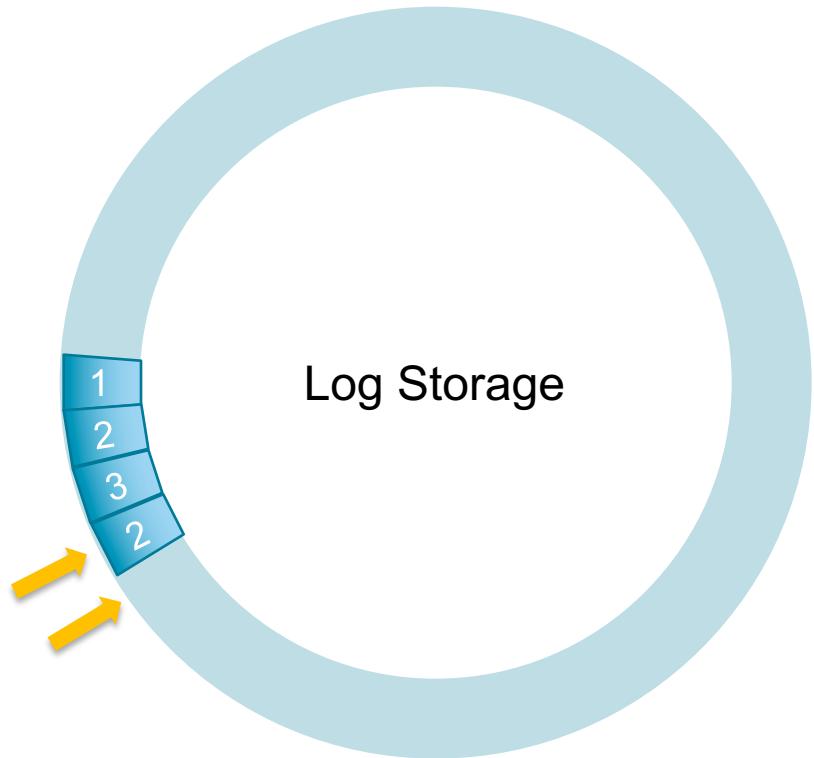
What is Log-Structured?

Record 3



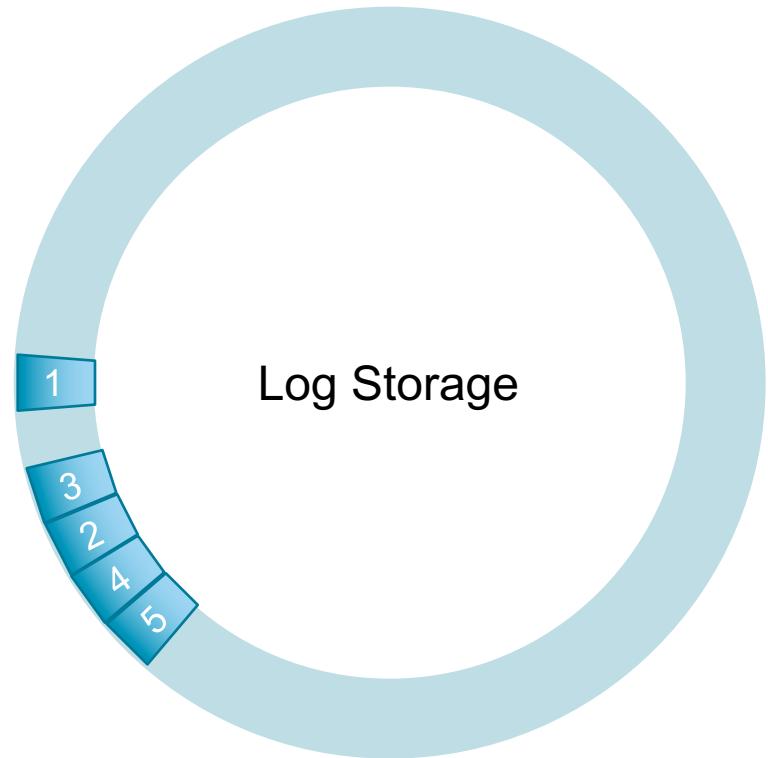
What is Log-Structured?

Record 2



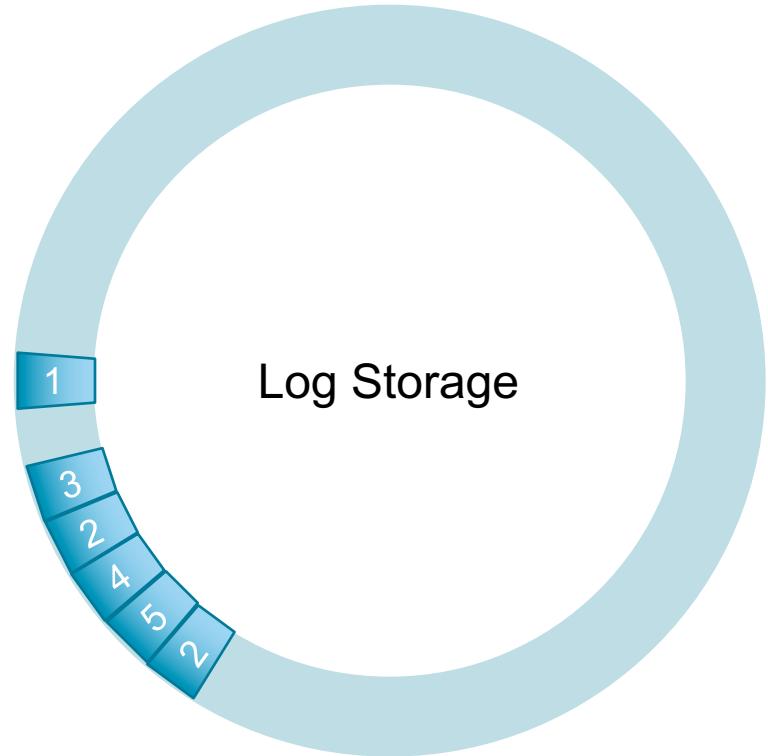
What is Log-Structured?

Record 5



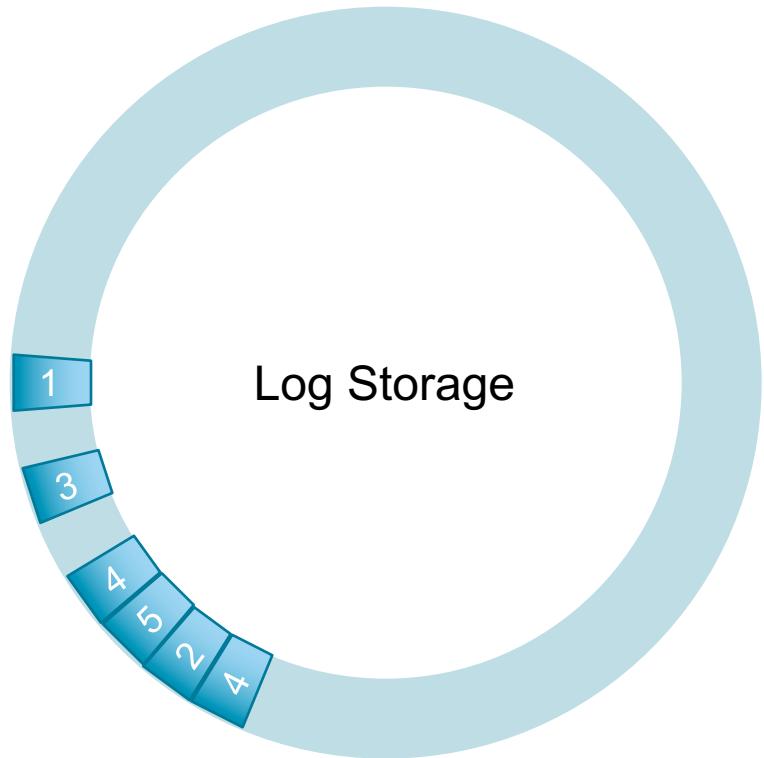
What is Log-Structured?

Record 2



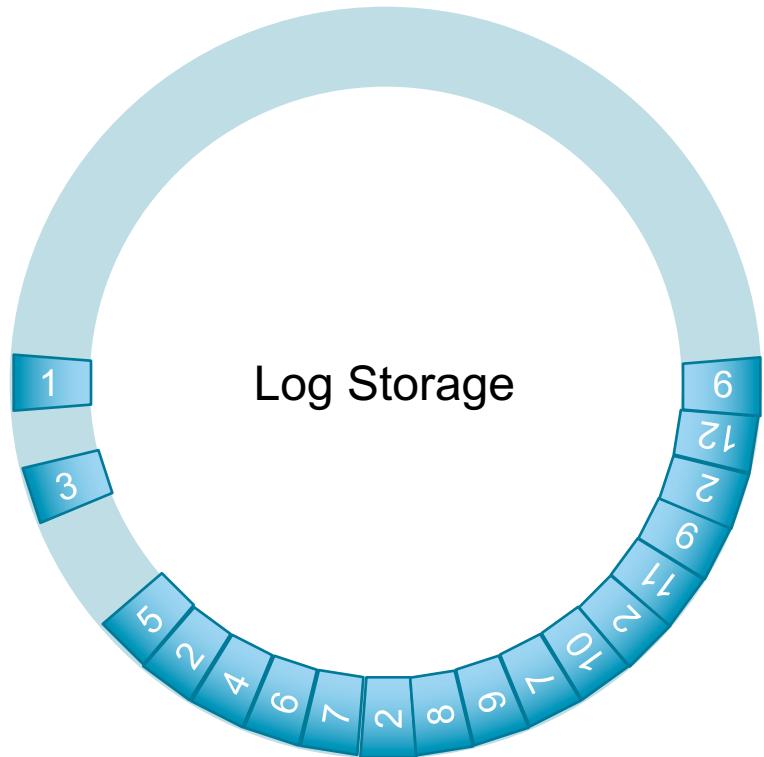
What is Log-Structured?

Record 4

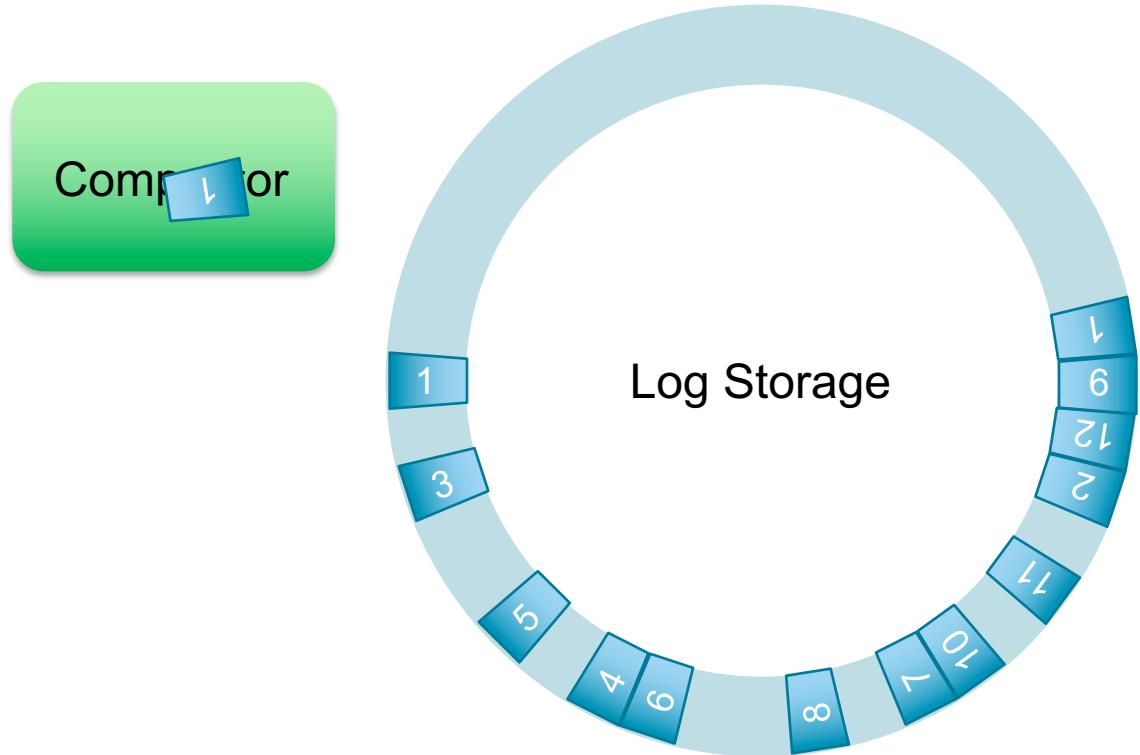


What is Log-Structured?

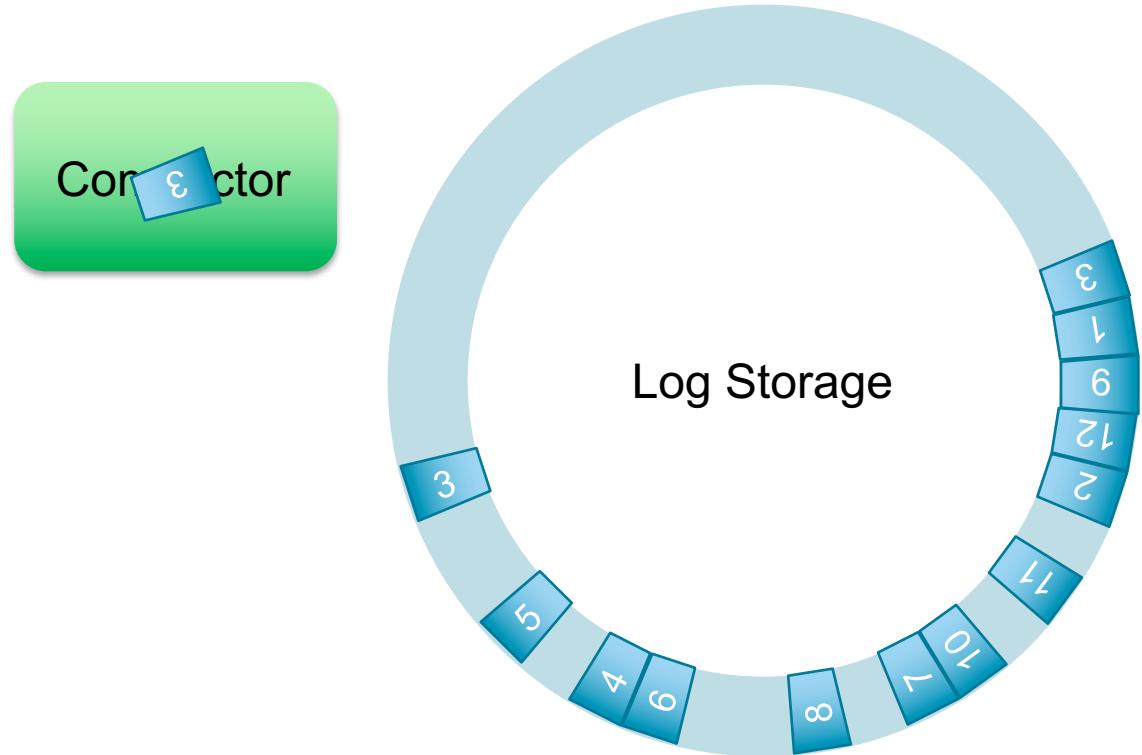
Record 9



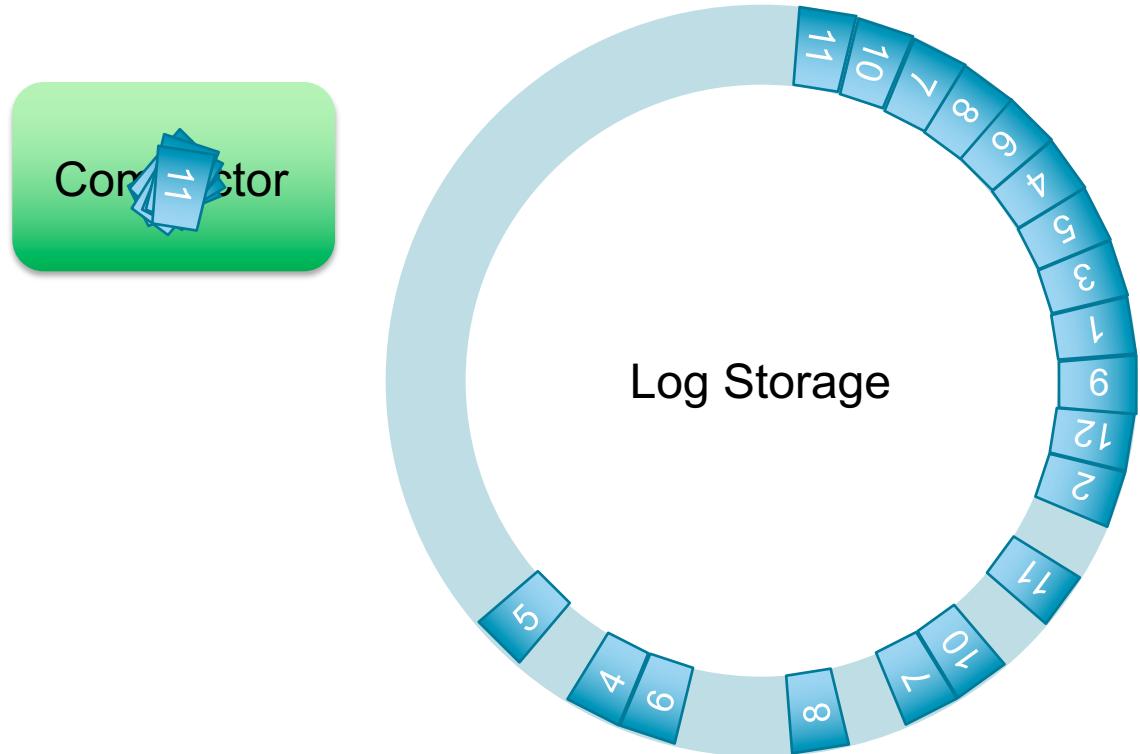
What is Log-Structured?



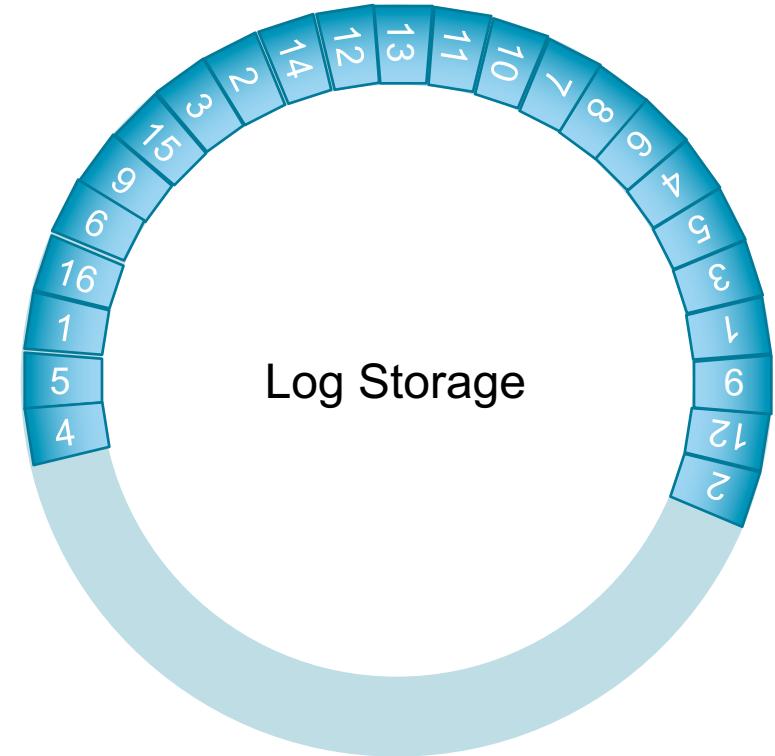
What is Log-Structured?



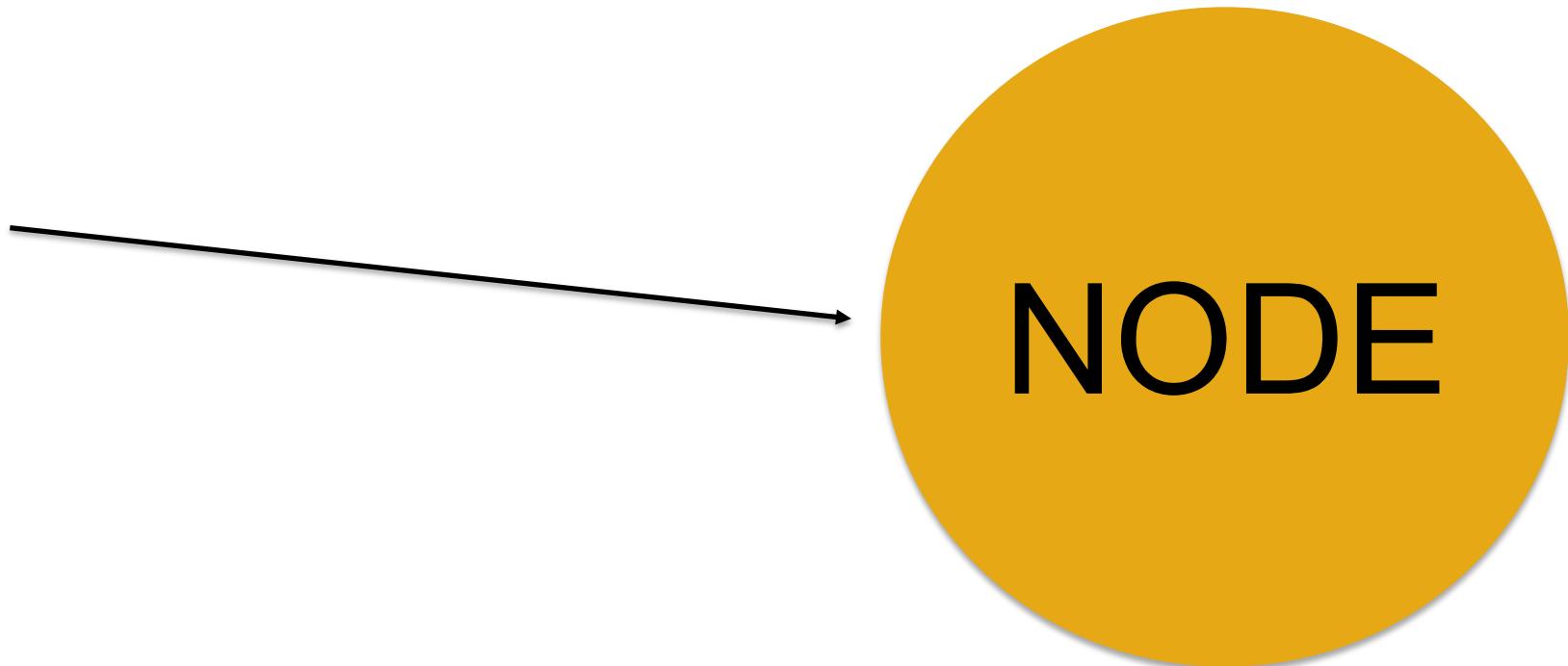
What is Log-Structured?



What is Log-Structured?



Cassandra's Write Path



MemTable →

RAM

← 2 ComeTo DSE TX Dallas

Commit Log →

HDD



4	IgotUr Data	TX	Austin
5	Always Onomnom	TX	Dallas
2	ComeTo DSE	TX	Dallas
4	Lone Star	TX	El Paso
5	Dev Awesome	TX	Houston
6	Lone Node	TX	Snyder

RAM

7	Data Rowman	TX	Austin
11	Prepar yer Query	TX	Austin
10	Cluster yer Rows	TX	Dallas
3	Lone Node	TX	Dallas
12	Learnin' to Model	TX	Houston
8	Lovin' Ur Bytes	TX	Sealy

HDD

SSTable
(immutable) →

4	IgotUr Data	TX	Austin
5	Always Onomnom	TX	Dallas
2	ComeTo DSE	TX	Dallas
4	Lone Star	TX	El Paso
5	Dev Awesome	TX	Houston
3	Lone Node	TX	Snyder

Cassandra's Read Path

Reading Data

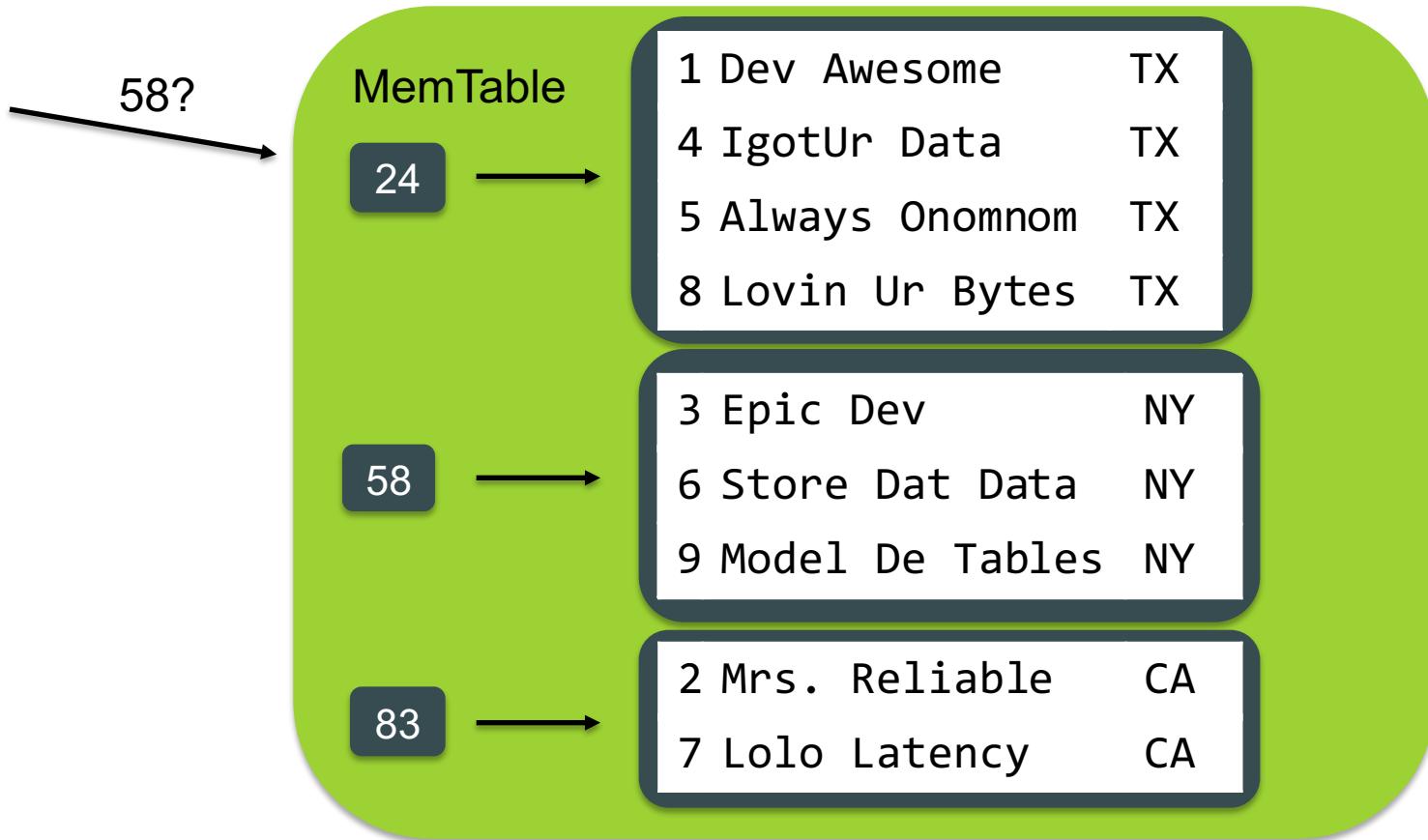
MemTable

SSTable #1

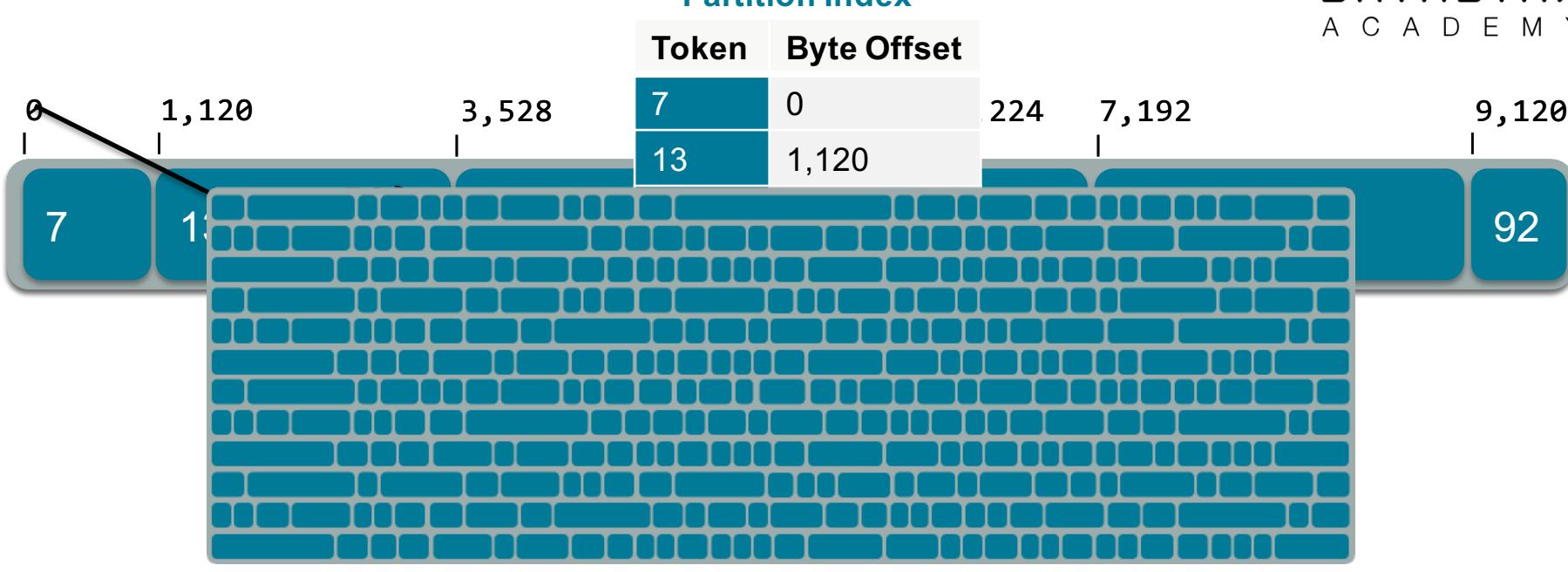
SSTable #2

SSTable #3

Reading a MemTable

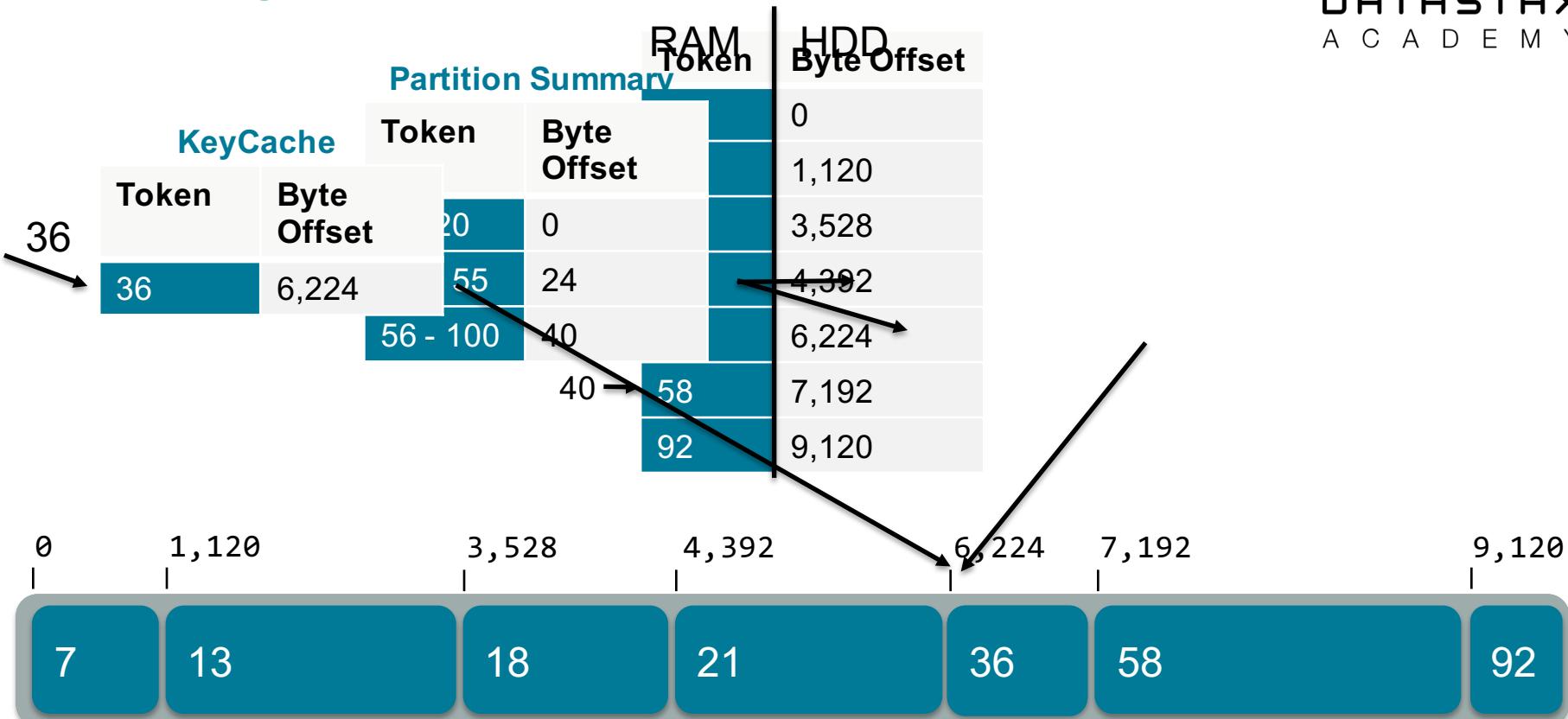


Reading an SSTable



58

Reading an SSTable



Bloom Filter



DataStax Enterprise 6.0

Read Path Optimizations

- No key cache
- Partition index changed to a trie-based data structure
- SSTable lookups in this format scream!
- Huge performance improvements; especially for large SSTables



- Migrating from OSS Apache Cassandra™ is seamless
- DataStax Enterprise can tell what kind of SSTable format it's working with
- As old SSTables are compacted, DataStax Enterprise writes them out in the new format

Loading Data into Cassandra

Loading Data into Cassandra

DSBulk – What is it?

- Moves Cassandra data to/from files in the file system
- Uses both CSV or JSON formats
- Command-line interface
- DSE Customer First feature

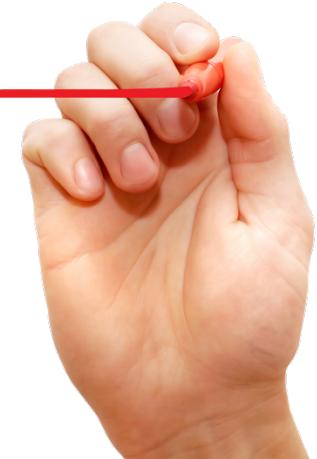


Loading Data into Cassandra

DSBulk – Why?

- Loading a lot of data into Cassandra has been difficult for a long time
- Unloading was needed too
- Previous tools were not ideal:
 - CQLSH COPY FROM is non-performant and not robust
 - SSTableLoader requires data to be in SSTable format
 - cassandra-loader is not formally supported

Why?



Loading Data into Cassandra

DSBulk Use-cases:

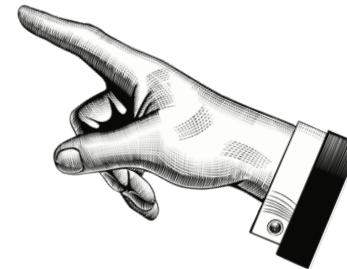
- Loading data from a pile of files
- One-time load or part of production flow
- Initial developer experience (load an existing familiar DB)
- Unload data for backup
- Migration from DSE to DSE (due to data model changes)

Loading Data into Cassandra

DSBulk Example

```
dsbulk load -url file1.csv -k ks1 -t table1
```

- Parameters:
 - file1.csv – this is the input file
 - ks1 – this is the keyspace name
 - table1 – this is the table name
- Steps:
 - Create the keyspace and table
 - Map the column using the header or via a config file
 - Run the command



Loading Data into Cassandra

DSBulk - Let's try it!

- Open a browser and go to <Your node's IP Address>:9091
- Click on the notebook “Core Cassandra: Data Loading”
- Work your way through the steps of this notebook (it's short)



Loading Data into Cassandra

Quick Review

- dsbulk is a command line tool for loading/unloading data
 - Use header or config file to map columns
 - Works for CSV and JSON
 - Create your table first
 - Handles various data types (e.g., text, date, float)



Data Availability

Cassandra Data Availability

What's it all about?

Availability = Replication

Replication implies consistency concerns

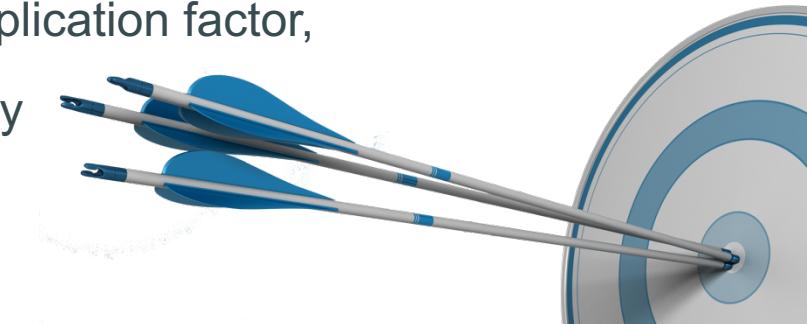
Consistency can be atomic



Cassandra Data Availability

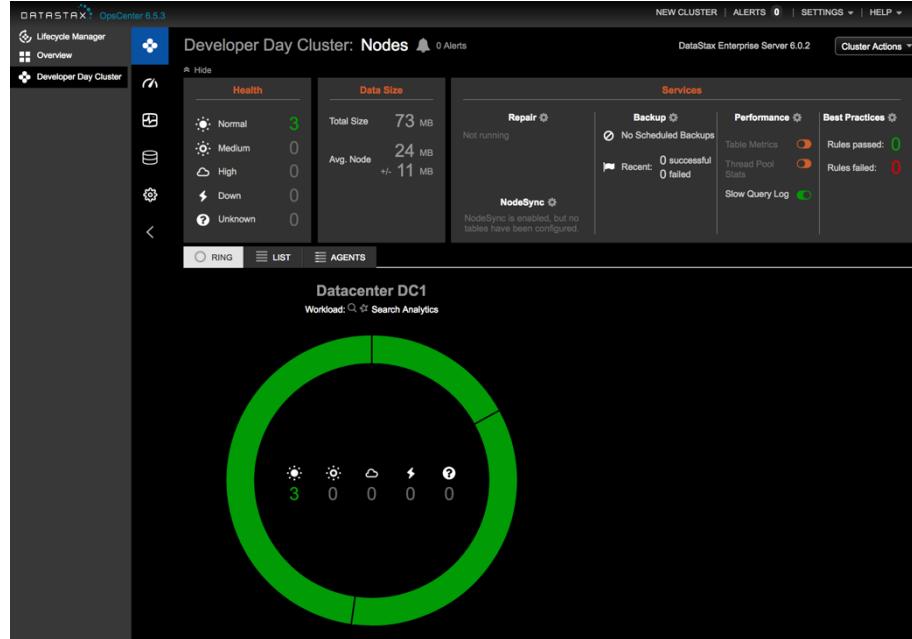
Some key concepts

- Replication Factor – the number of copies of your data
 - Increasing replication increases chances of availability
 - Increasing replication increases chances of inconsistency
- Consistency Level – the number of acknowledged copies read/written
- If number of writes + number of reads > replication factor,
 - You are guaranteed strong consistency



OpsCenter

For Creating, Managing and Monitoring Your Cluster



Cassandra Data Availability

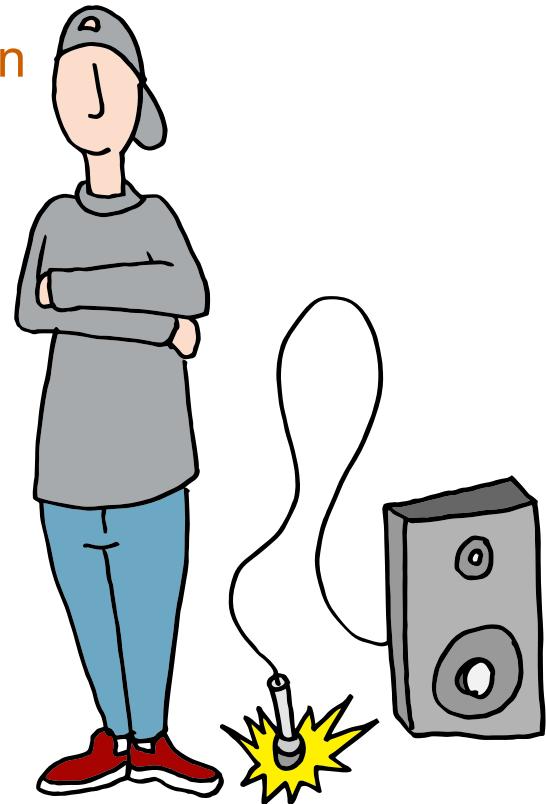
“Core Cassandra: Data Availability” Notebook - Let’s try it!



Final Words

Thanks for Participating in Cassandra Core Session

- Now, you should:
 - Understand what Apache Cassandra™ is
 - Know what Cassandra does
 - Yearn to have Cassandra in your shop
- Want More?
 - Visit academy.datastax.com
 - It's free!





The End