MongoDB Ops Manager Monitoring Metrics

# Introduction

MongoDB Ops Manager offers detailed monitoring of MongoDB clusters, allowing users to observe system health, database performance, and replication behaviors in real-time. This document outlines all the key metrics available in MongoDB Ops Manager and explains their significance in managing and troubleshooting your MongoDB deployments.

# 1. Host Metrics

## 1.1 CPU Usage

Description: Monitors the percentage of CPU resources consumed by the MongoDB processes.

Threshold: High CPU usage over 90% for sustained periods can impact query and write performance.

## 1.2 Memory Usage

Description: Tracks the amount of system memory being utilized by MongoDB, ensuring efficient use of resources.

Threshold: Memory usage exceeding 80% may indicate that scaling resources are necessary.

## 1.3 Disk I/O

Description: Measures the I/O operations per second (IOPS) on the disk used by MongoDB.

Use Case: Identifies disk bottlenecks, especially during heavy write loads, which can affect performance.

## 1.4 Network Traffic

Description: Reports on the volume of data sent and received by MongoDB across the network.

Use Case: Helps ensure that network capacity can handle MongoDB’s replication and sharding traffic.

# 2. Database Metrics

## 2.1 OpCounters (Operation Counters)

Description: Tracks the number of operations (insert, update, delete, query) MongoDB handles.

Use Case: Helps identify trends in database activity and balance read/write loads.

## 2.2 Lock Percentage

Description: Indicates the percentage of time the database is locked, affecting other operations.

Threshold: High lock percentages can significantly degrade performance; locking above 30% may need optimization.

## 2.3 Connections

Description: Monitors the number of active client connections to the MongoDB server.

Threshold: A high number of connections (near or over the connection pool limit) can lead to resource exhaustion.

## 2.4 Page Faults

Description: Occurs when MongoDB must retrieve data from disk rather than from memory.

Use Case: High page faults indicate insufficient memory allocation, which affects read performance.

## 2.5 Cache Usage

Description: Tracks how much of MongoDB's data is cached in memory.

Use Case: Ensures efficient memory usage, improving read performance by keeping frequently accessed data in RAM.

# 3. Replication Metrics

## 3.1 Replication Lag

Description: Measures the time delay between the primary node writing an operation and the secondary node replicating it.

Threshold: Lag over 10 seconds may indicate replication issues or bottlenecks.

## 3.2 Oplog Window

Description: The period of time that the oplog can hold changes for replication.

Use Case: Ensures that the oplog size is appropriate to handle workload peaks without losing data during replication.

## 3.3 Rollback Count

Description: Monitors the number of rollback operations triggered during failovers.

Use Case: Frequent rollbacks may indicate replication instability and a need for investigation.

# 4. Shard Metrics

## 4.1 Chunk Distribution

Description: Monitors the number and size of data chunks across shards.

Use Case: Ensures data is evenly distributed across shards for balanced query load.

## 4.2 Balancer Operations

Description: Tracks the activity of the balancer, responsible for moving chunks between shards.

Use Case: Imbalances in chunk distribution can cause inefficient queries and require rebalancing.

## 4.3 Jumbo Chunks

Description: Identifies chunks that are too large to move during rebalancing.

Use Case: Jumbo chunks can prevent the balancer from achieving optimal distribution. Manual intervention may be needed to split or manage them.

# 5. Index Metrics

## 5.1 Index Miss Ratio

Description: The percentage of queries that fail to use an index.

Threshold: High index miss ratios (above 20%) may lead to slow queries and should prompt index optimization.

## 5.2 Index Build Time

Description: Measures the time MongoDB takes to build or rebuild an index.

Use Case: Long index build times can degrade performance and affect availability during maintenance.

## 5.3 Index Size

Description: Reports the total size of all indexes on a collection or database.

Use Case: Helps ensure indexes are appropriately sized for the underlying dataset and query patterns.

# 6. Write Performance Metrics

## 6.1 Write Throughput

Description: Measures the number of write operations per second handled by the MongoDB server.

Use Case: A sudden drop in write throughput can indicate network, disk, or replication issues.

## 6.2 Journal Commit Interval

Description: The time it takes for MongoDB to commit changes to the journal.

Use Case: High commit intervals may indicate storage performance bottlenecks.

## 6.3 Write Concern Time

Description: The time it takes for a write operation to meet the write concern (e.g., majority acknowledgment).

Use Case: Inconsistent or long write concern times may indicate issues with replication or disk I/O.

# 7. Query Performance Metrics

## 7.1 Query Execution Time

Description: The time it takes MongoDB to execute a query.

Use Case: Long query times can indicate the need for better indexes or query optimization.

## 7.2 Query Planner Stats

Description: Provides details on the query planner's choice of index and execution path.

Use Case: Helps identify and optimize inefficient query plans.

## 7.3 Scan and Order

Description: Tracks queries that require scanning documents in an unoptimized order.

Use Case: Frequent scan and order operations may require additional indexes or query optimization.

# 8. Alerting and Thresholds

## 8.1 Configuring Alerts

Description: MongoDB Ops Manager allows users to configure custom alerts for various thresholds.

Common Alerts:

- CPU usage over 85% for sustained periods.

- Replication lag greater than 10 seconds.

- Disk space usage exceeding 90%.

- Connection pool exhaustion (e.g., above 80% of max connections).

# 9. Backup and Recovery Metrics

## 9.1 Backup Success Rate

Description: Tracks the success rate of scheduled MongoDB backups.

Use Case: Ensures backups are completed successfully without errors.

## 9.2 Restore Time

Description: Measures the time taken to restore a backup.

Use Case: Ensures that restore times are within acceptable disaster recovery thresholds.

# Conclusion

By tracking the metrics above, you can gain deep insights into the performance and health of your MongoDB deployment. MongoDB Ops Manager provides a flexible monitoring platform that enables proactive management, alerting, and troubleshooting of MongoDB clusters, helping to maintain optimal performance and availability.