

# Understanding Consistency Models and Capacity Units

---

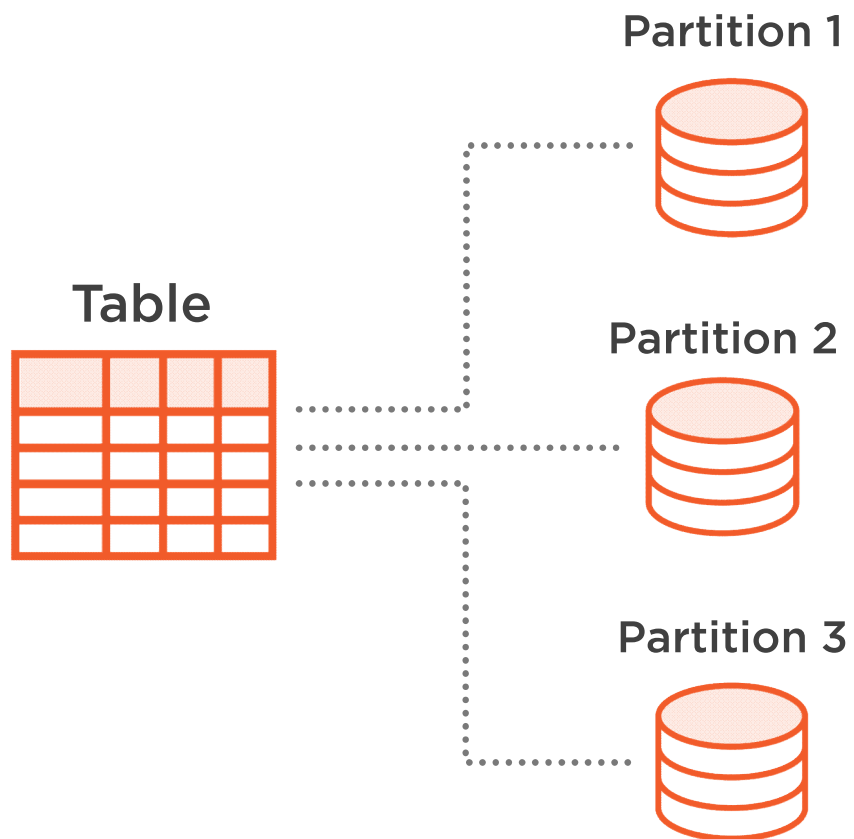


**Abhaya Chauhan**

@AbhayaChauhan [www.abhayachauhan.com](http://www.abhayachauhan.com)



# What Is a Partition?



# Benefits of Partitioning

**Manageability**

**Performance**

**Availability**



## Vertical Partitioning

Partition 1

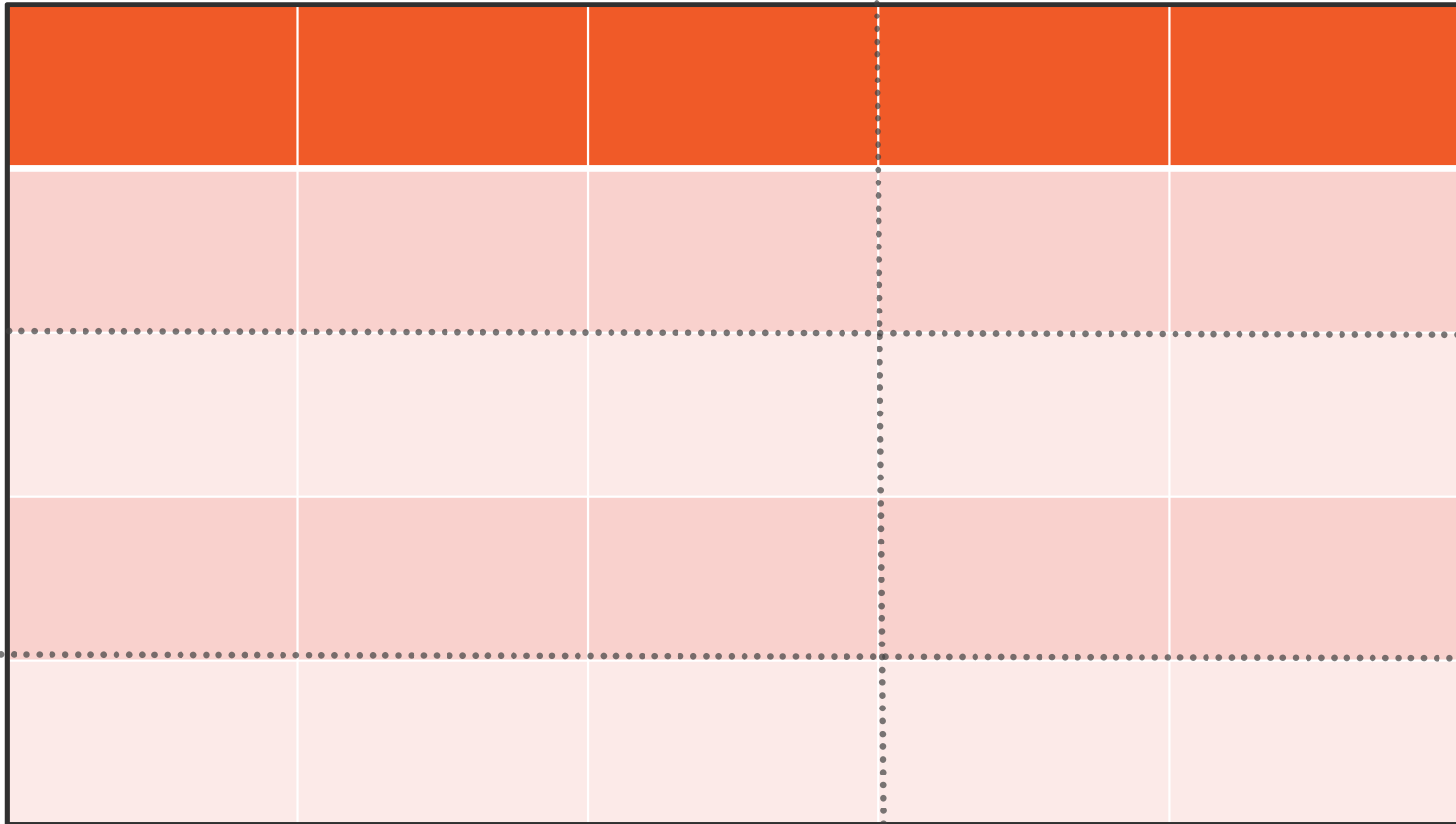
Partition 2

Horizontal  
Partitioning

Partition 1

Partition 2

Partition 3



Horizontal  
Partitioning

Partition 1

ABC

Partition 2

XYZ

XYZ

Partition 3

DEF

Partition Key				
ABC				
XYZ				
XYZ				
DEF				



# Types of Queries

## **Known Partition Key**

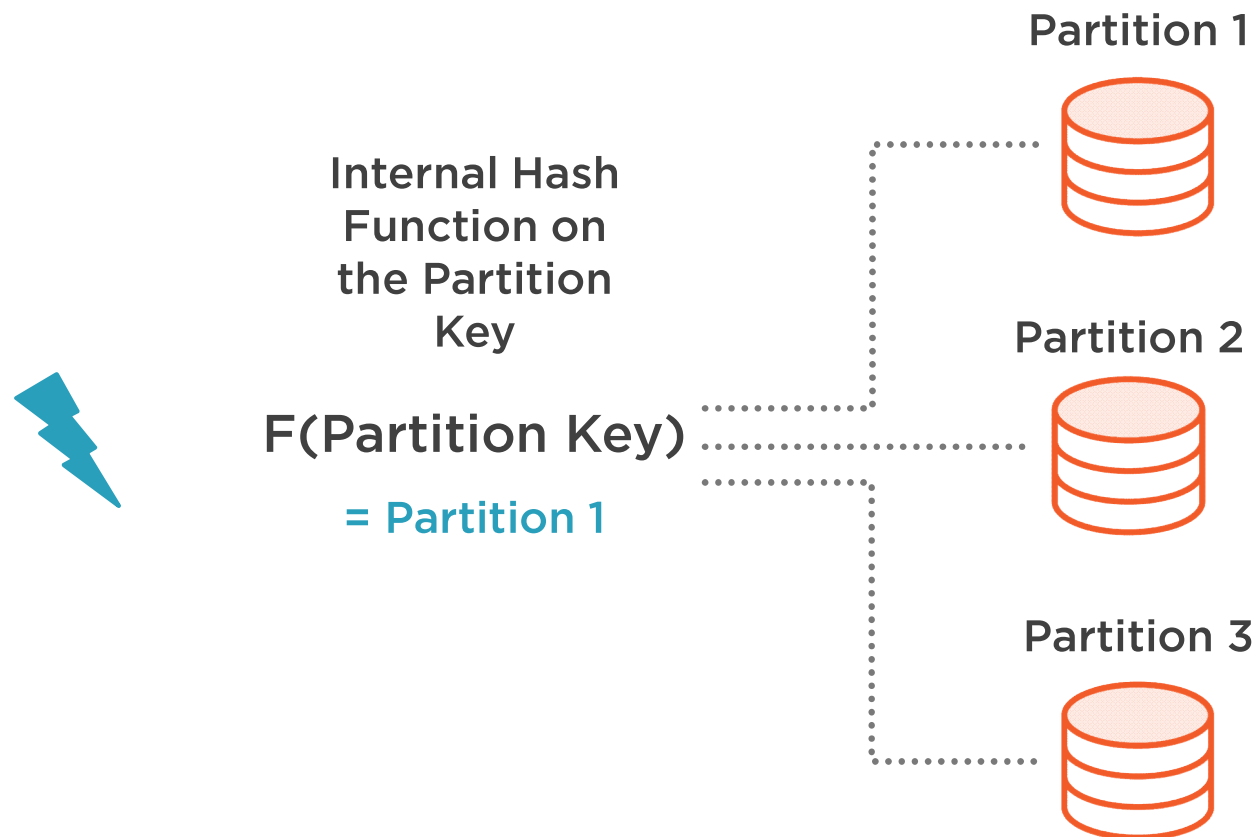
- DynamoDB knows which partition to access

## **Unknown Partition Key**

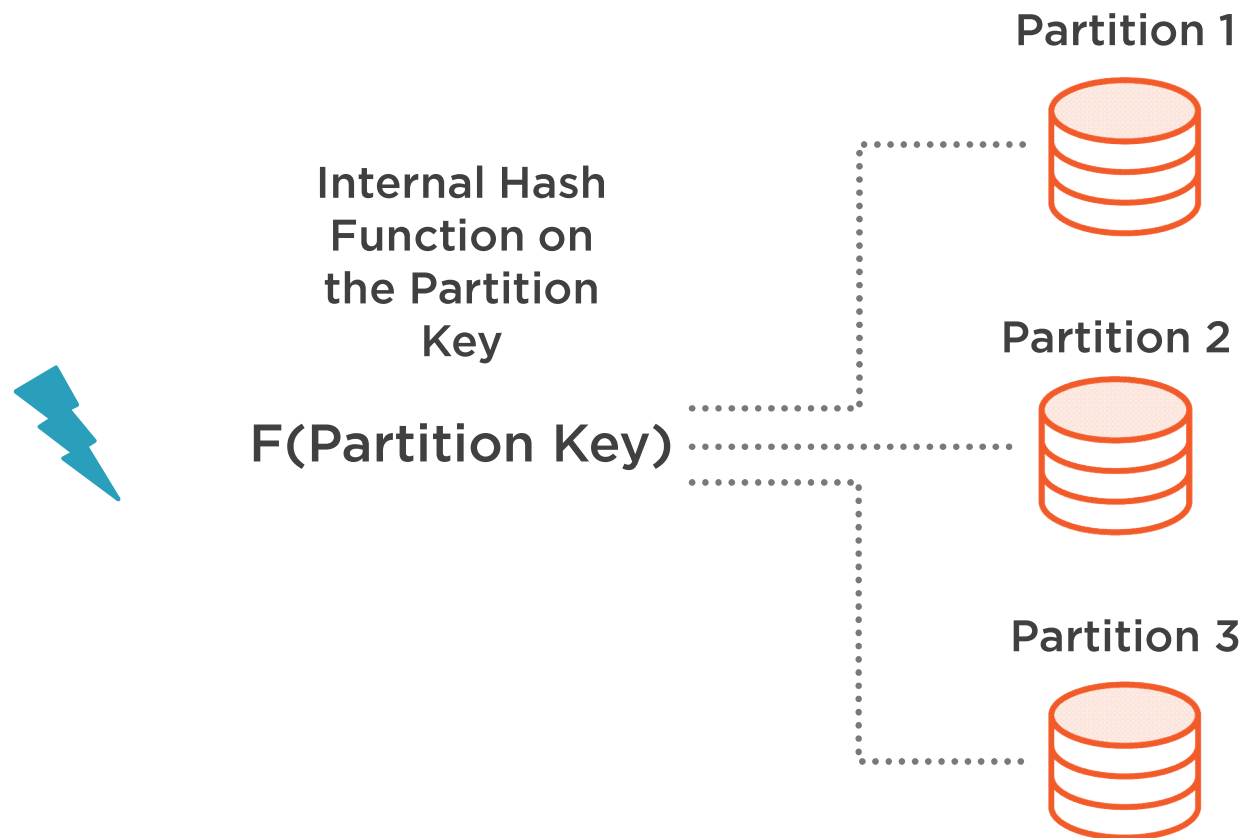
- DynamoDB doesn't know which partition to access



# Known Partition Key



# Unknown Partition Key





# Consistency Models

**Strongly Consistent**

**Eventually Consistent**



# Consistency Models

## Strongly Consistent

Reads return latest results

## Eventually Consistent



# Consistency Models

## Strongly Consistent

Reads return latest results

## Eventually Consistent

Reads may not return latest results



# Consistency Models

## Strongly Consistent

Reads return latest results

Consumes more RCUs

## Eventually Consistent

Reads may not return latest results



# Consistency Models

## Strongly Consistent

Reads return latest results

Consumes more RCUs

## Eventually Consistent

Reads may not return latest results

Consumes half RCUs of Consistent Read



# Consistency Models

## Strongly Consistent

Reads return latest results

Consumes more RCUs

Higher read latency

## Eventually Consistent

Reads may not return latest results

Consumes half RCUs of Consistent Read



# Consistency Models

## Strongly Consistent

Reads return latest results

Consumes more RCUs

Higher read latency

## Eventually Consistent

Reads may not return latest results

Consumes half RCUs of Consistent Read

Lower read latency



# Consistency Models

## Strongly Consistent

Reads return latest results

Consumes more RCUs

Higher read latency

Not default

## Eventually Consistent

Reads may not return latest results

Consumes half RCUs of Consistent Read

Lower read latency





# Consistency Models

## Strongly Consistent

Reads return latest results

Consumes more RCUs

Higher read latency

Not default

## Eventually Consistent

Reads may not return latest results

Consumes half RCUs of Consistent Read

Lower read latency

Default Read Model



# Capacity Units

**Predefined throughput for a Table**

**DynamoDB will provision for your requirements**

**Specify in terms of:**

- Read Capacity (RCUs)
- Write Capacity (WCUs)



# Write Capacity Units

**1 WCU = up to 1KB of data / second**

- Each write is rounded up to the closest 1KB

**Formula:**

- Round up ( Item size / 1KB )

**Examples**

- Write 5KB of data
  - Round up ( 5KB / 1KB ) = 5 WCUs
- Write 500 bytes
  - Round up ( 0.5KB / 1KB ) = 1 WCU



# Read Capacity Units

## Two modes of reading

- Strongly Consistent
- Eventually Consistent

## Strongly Consistent:

- 1 RCU = up to 4KB of data / second

## Eventually Consistent

- 0.5 RCU = up to 4KB of data / second

**Each read is rounded up to the closest 4KB**



# Read Capacity Units

## Formulas

- Strongly Consistent
  - Round up ( Item size / 4KB )
- Eventually Consistent
  - Round up ( Item size / 4KB ) / 2



# Read Capacity Units

## Examples

- Read 20KB of data
  - Strongly consistent mode
    - Round up  $(20\text{KB} / 4\text{KB}) = 5$  RCUs
  - Eventually consistent mode
    - Round up  $(20\text{KB} / 4\text{KB}) / 2 = 2.5$  RCUs
- Read 2KB
  - Strongly consistent mode
    - Round up  $(2\text{KB} / 4\text{KB}) = 1$  RCU
  - Eventually consistent mode
    - Round up  $(2\text{KB} / 4\text{KB}) / 2 = 0.5$  RCU



# Summary

