AWS DynamoDB Throughput Capacity

# AWS DynamoDB Throughput Capacity

* AWS DynamoDB throughput capacity depends on the read/write capacity modes for processing reads and writes on the tables.
* DynamoDB supports two types of read/write capacity modes:  
  + On-demand
  + Provisioned

**NOTE – Provisioned mode is covered in the AWS Certified Developer – Associate exam (DVA-C01) esp. the calculations. On-demand capacity mode is latest enhancement and does not yet feature in the exams.**

## Provisioned Mode

* Provisioned mode requires you to specify the number of reads and writes per second as required by the application
* Provisioned throughput is the maximum amount of capacity that an application can consume from a table or index
* If the provisioned throughput capacity on a table or index is exceeded, it is subject to request throttling
* Provisioned mode is a good for applications
  + predictable application traffic
  + consistent traffic
  + ability to forecast capacity requirements to control costs
* Provisioned mode provides the following capacity units
  + Read Capacity Units (RCU)
    - Total number of read capacity units required depends on the item size, and the consistent read model (eventually or strongly)
    - one RCU represents
      * **two eventually consistent reads per second, for an item up to 4 KB in size i.e. 8 KB**
      * **one strongly consistent read per second for an item up to 4 KB in size** i.e. 2x cost of eventually consistent reads
      * **Transactional read requests require two read capacity units to perform one read per second for items up to 4 KB.**i.e. 2x cost of strongly consistent reads
    - DynamoDB must consume additional read capacity units for items greater than 4 KB for e.g. for an 8 KB item size, 2 read capacity units to sustain one strongly consistent read per second, 1 read capacity unit if you choose eventually consistent reads, or 4 read capacity units for a transactional read request would be required
    - Item size is rounded off to 4 KB equivalents for e.g. a 6 KB or a 8 KB item in size would require the same RCU
  + Write Capacity Units (WCU)
    - Total number of write capacity units required depends on the item size only
    - **one write per second for an item up to 1 KB in size**
    - **Transactional write requests require 2 write capacity units to perform one write per second for items up to 1 KB.**i.e. 2x cost of general write.
    - DynamoDB must consume additional read capacity units for items greater than 1 KB for an 2 KB item size,  2 write capacity units would be required to sustain one write request per second or 4 write capacity units for a transactional write request
    - Item size is rounded off to 1 KB equivalents for e.g. a 0.5 KB or a 1 KB item would need the same WCU

### Provisioned Mode Examples

* DynamoDB table with provisioned capacity of 10 RCUs and 10 WCUs can support
  + Read throughput
    - Eventual consistency = 4KB \* 10 \* 2 = 80KB/sec
    - Strong consistency = 4KB \* 10 = 40KB/sec
    - Transactional consistency = 4KB \* 10 \* 1/2 = 20KB/sec
  + Write throughput
    - Eventual and Strong consistency = 10 \* 1KB = 10KB/sec
    - Transaction consistency = 10 \* 1KB \* 1/2 = 5KB/sec
* Capacity units required for reading and writing 15KB item
  + Read capacity units – 15KB rounded to 4 blocks of 4KB = 4 RCUs
    - Eventual consistency 4 RCUs \* 1/2 = 2 RCUs
    - Strong consistency 4 RCUs \* 1 = 4 RCUs
    - Transactional consistency 4 RCUs \* 2 = 8 RCUs
  + Write capacity units 15KB = 15 WCUs
    - Eventual and Strong consistency 15 WCUs \* 1 = 15 WCUs
    - Transactional consistency 15 WCUs \* 2 = 30 RCUs

## On-demand Mode

* On-demand mode provides flexible billing option capable of serving thousands of requests per second without capacity planning
* There is no need to specify the expected read and write throughput
* Charged for only the reads and writes that the application performs on the tables in terms of read request units and write request units.
* Offers pay-per-request pricing for read and write requests so that you pay only for what you use
* DynamoDB adapts rapidly to accommodate the changing load
* DynamoDB on-demand using Request units which are similar to provisioned capacity Units
* On-demand mode does not support reserved capacity

## DynamoDB Burst Capacity

* DynamoDB provides some flexibility in the per-partition throughput provisioning by providing burst capacity.
* If partition’s throughput is not fully used, DynamoDB reserves a portion of that unused capacity for later bursts of throughput to handle usage spikes
* DynamoDB currently retains up to 5 minutes (300 seconds) of unused read and write capacity.
* During an occasional burst of read or write activity, these extra capacity units can be consumed quickly—even faster than the per-second provisioned throughput capacity that you’ve defined for your table.
* DynamoDB can also consume burst capacity for background maintenance and other tasks without prior notice.

## DynamoDB Adaptive Capacity

* Adaptive capacity is a feature that enables DynamoDB to run imbalanced workloads indefinitely.
* DynamoDB distributes the data across partitions and the throughput capacity is distributed equally across these partitions
* However, when data access is imbalanced, a “hot” partition can receive a higher volume of read and write traffic compared to other partitions leading to throttling errors on that partition.
* DynamoDB adaptive capacity enables the application to continue reading and writing to hot partitions without being throttled, provided that traffic does not exceed the table’s total provisioned capacity or the partition maximum capacity.
* It minimizes throttling due to throughput exceptions.
* It also helps reduce costs by enabling provisioning of only the needed throughput capacity.
* Adaptive capacity is enabled automatically for every DynamoDB table, at no additional cost.

## DynamoDB Throttling

* Each partition on a DynamoDB table is subject to a hard limit of 1,000 write capacity units and 3,000 read capacity units.
* If the workload is unevenly distributed across partitions, or if the workload relies on short periods of time with high usage (a burst of read or write activity), the table might be throttled.
  + **Distribute read and write operations as evenly as possible across your table.** A hot partition can degrade the overall performance of your table.
  + **Implement a caching solution**. If the workload is mostly read access to static data, then query results can be delivered much faster if the data is in a well‑designed cache rather than in a database. DynamoDB Accelerator (DAX) is a caching service that offers fast in‑memory performance for your application. ElastiCache can be used as well.
  + **Implement error retries and exponential backoff.** Exponential backoff can improve an application’s reliability by using progressively longer waits between retries. If using an AWS SDK, this logic is built‑in.

[do\_widget id=text-15]

## AWS Certification Exam Practice Questions

* Questions are collected from Internet and the answers are marked as per my knowledge and understanding (which might differ with yours).
* AWS services are updated everyday and both the answers and questions might be outdated soon, so research accordingly.
* AWS exam questions are not updated to keep up the pace with AWS updates, so even if the underlying feature has changed the question might not be updated
* Open to further feedback, discussion and correction.

[do\_widget id=text-15]

1. You need to migrate 10 million records in one hour into DynamoDB. All records are 1.5KB in size. The data is evenly distributed across the partition key. How many write capacity units should you provision during this batch load?
   1. 6667
   2. 4166
   3. **5556**( 2 write units (1 for each 1KB) \* 10 million/3600 secs)
   4. 2778
2. A meteorological system monitors 600 temperature gauges, obtaining temperature samples every minute and saving each sample to a DynamoDB table. Each sample involves writing 1K of data and the writes are evenly distributed over time. How much write throughput is required for the target table?
   1. 1 write capacity unit
   2. **10 write capacity units**( 1 write unit for 1K \* 600 gauges/60 secs)
   3. 60 write capacity units
   4. 600 write capacity units
   5. 3600 write capacity units
3. A company is building a system to collect sensor data from its 36000 trucks, which is stored in DynamoDB. The trucks emit 1KB of data once every hour. How much write throughput is required for the target table. Choose an answer from the options below
   1. **10**
   2. 60
   3. 600
   4. 150
4. A company is using DynamoDB to design storage for their IOT project to store sensor data. Which combination would give the highest throughput?
   1. **5 Eventual Consistent reads capacity with Item Size of 4KB** (40KB/s)
   2. 15 Eventual Consistent reads capacity with Item Size of 1KB (30KB/s)
   3. 5 Strongly Consistent reads capacity with Item Size of 4KB (20KB/s)
   4. 15 Strongly Consistent reads capacity with Item Size of 1KB (15KB/s)
5. If your table item’s size is 3KB and you want to have 90 strongly consistent reads per second, how many read capacity units will you need to provision on the table? Choose the correct answer from the options below
   1. **90**
   2. 45
   3. 10
   4. 19

## References