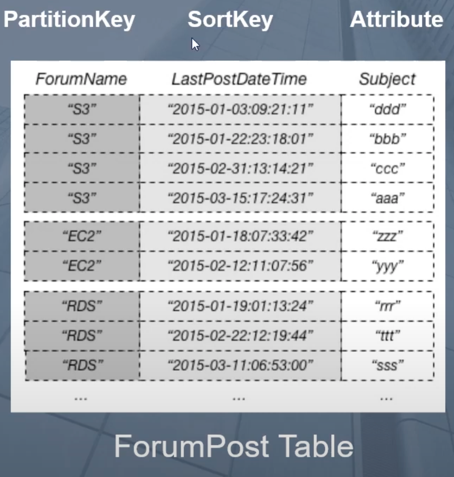
* **NoSQL & SQL**
  + Collection (table)
  + Document (row)
  + Key (field)

|  |  |
| --- | --- |
| Understanding the differences between Azure Cosmos DB NoSQL and relational  databases | Microsoft Docs | The basics of NoSQL databases — and why we need them |

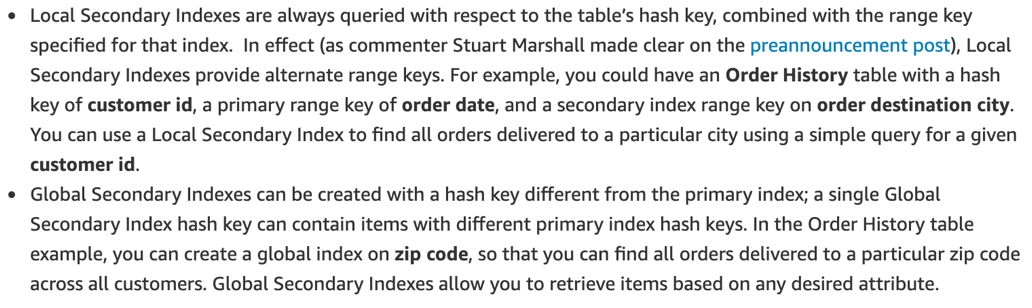
* **DynamoDB**
* Là NoSQL Database của AWS
* Đặc điểm
  + Fast & flexiable
  + Data được lưu trên SSD
  + Chậm hơn MongoDB, cộng đồng dynamoDB cũng bé hơn, dynamoDB chỉ dùng trong AWS
  + Spread across 3 geographically distinct data centres
* Usecase
  + Thường dùng trong các trang thương mại điện tử
  + Lưu session state
* Price
  + <https://aws.amazon.com/dynamodb/pricing/>
* **Các thành phần**
  + **Table**
    - Hoạt động trên 1 region
    - Bao gồm:
      * Partition key & Sort key
      * Secondary indexes
      * Query/Scan/PartiQL
      * Read/write consistency
      * Read/write capacity
      * Global table
      * Conditional writes
      * Expiring Items By Time to Live (TTL)
      * Backup – Restore
      * Export S3
      * Encryption
      * Monitoring
  + **Item, Attribute**
    - Mỗi item là 1 list các attribute
    - 1 attribute có thể là
      * Partition key
      * Sort key
      * Attribute
    - Datatype: String, Number, Boolean, Binary, Null, List, Map, …



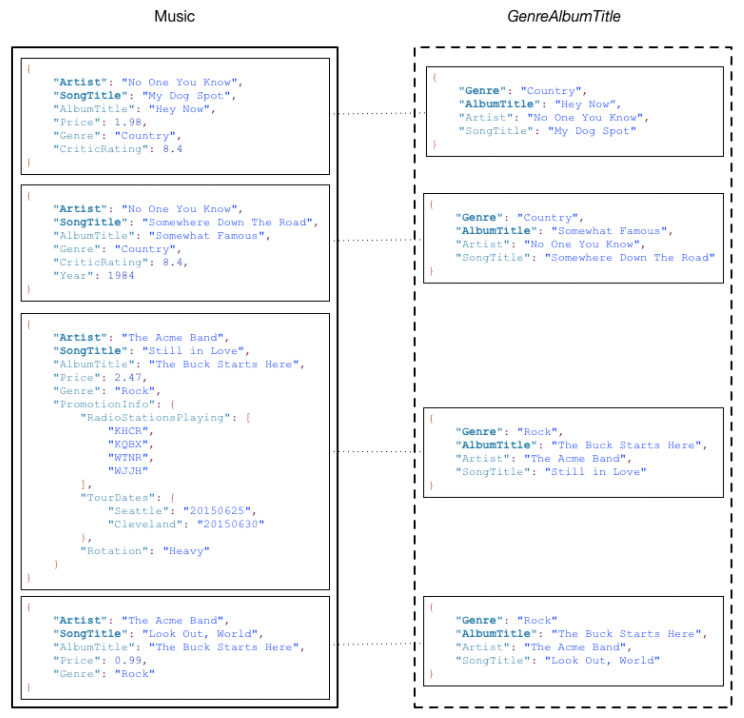
* + **Partition key & Sort key**
    - Partition key để khi query xác định duy nhất 1 item (Giống Primary key)
      * Giống group JSON, table theo attribute Partition key
      * Khi update item trong Partition key thì sẽ tự update table gốc
    - Sort key để sort hoặc search item có cùng Partition key (Optional)

|  |  |
| --- | --- |
| *https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/images/HowItWorksPeople.png*  *Partition key - PersonID* | *https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/images/HowItWorksMusic.png*  *Partition key – Artist*  *Sort key - SongTitle* |

* + **Secondary index**
    - Ngoài query theo Partition key thì có thể tự tạo ra các secondary index để query (Flexible hơn)
      * Giống nhóm Json, table theo attribute secondary index
      * Khi update item trong secondary index thì sẽ tự update table gốc
    - Có thể tạo nhiều index
    - Có 2 loại
      * Global secondary index
      * Local secondary index



[*https://aws.amazon.com/blogs/aws/now-available-global-secondary-indexes-for-amazon-dynamodb/*](https://aws.amazon.com/blogs/aws/now-available-global-secondary-indexes-for-amazon-dynamodb/)



* + **Query**
    - Query item dựa trên partition key
    - Có thể query, sort, filter, paging, count, …
  + **Scan**
    - Query item trên table hoặc index
    - <https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/Scan.html>
  + **PartiQL** 
    - Cung cấp câu query giống như thao tác với SQL (SQL-compatible)
    - Các câu lệnh support
      * Select(where, orderby, subquery)
      * Insert, Update, Delete
      * Transaction
      * Export result ra file CSV
    - Tools: NoSQL Workbench
  + **Capacity unit** (provisioned throughput capacity)
    - Là đơn vị đo request khối lượng request vào DB
    - Có 2 loại
      * Read capacity unit
      * Write capacity unit
  + **Read/write Consistency**
    - Read consistency
      * Eventual Consistent (Default)

Do data được replica giữa các zone 🡺 có thể đọc data cũ

* + - * Strongly Consistent

Đảm bảo luôn luôn đọc được data mới nhất

Đắt hơn

* + - * Transactional
    - Write consistency
      * Standard
      * Transactional
  + **Read/write capacity mode**
    - Config mode thanh toán

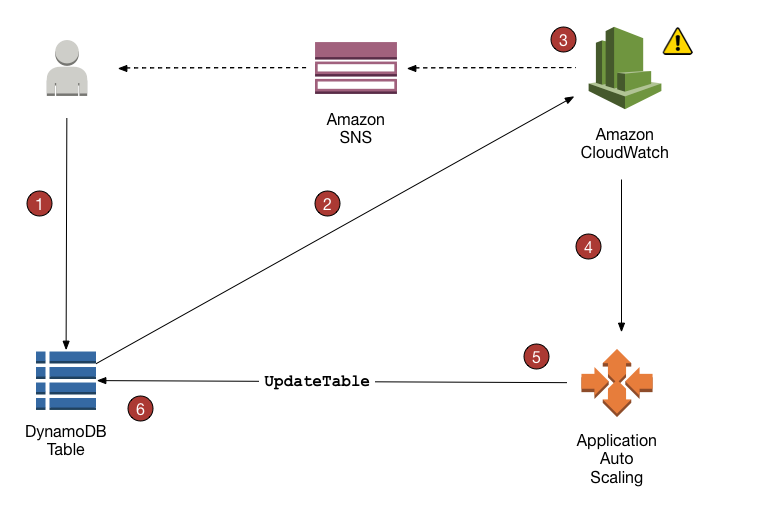
VD:

50 read/second, 50 write/second, mỗi item có size trung bình = 10 KB

Read consistency = Eventually consistent, Write consistency = Standard

🡺 Read capacity unit = 75 & Write capacity unit = 500

* + - Có 2 loại
      * On-demand
        + Thanh toán theo read,write/second capaciy
      * Provisioned (default, free-tier eligible)
        + Fix read/write capacity unit hoặc set auto scaling theo workload

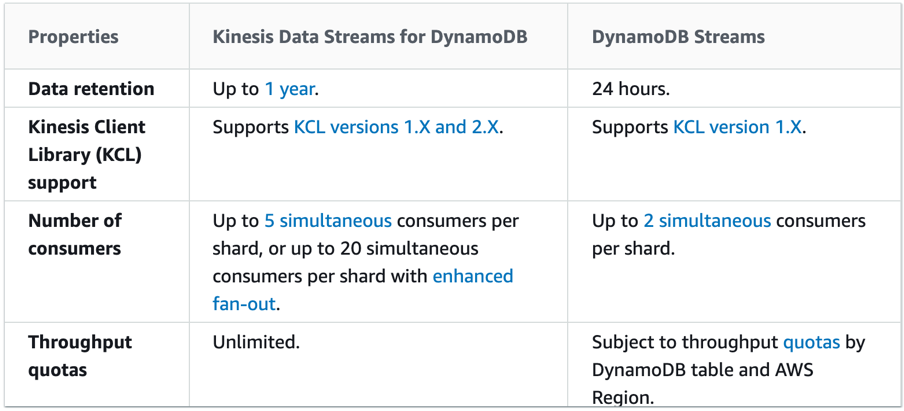


*Autoscaling*

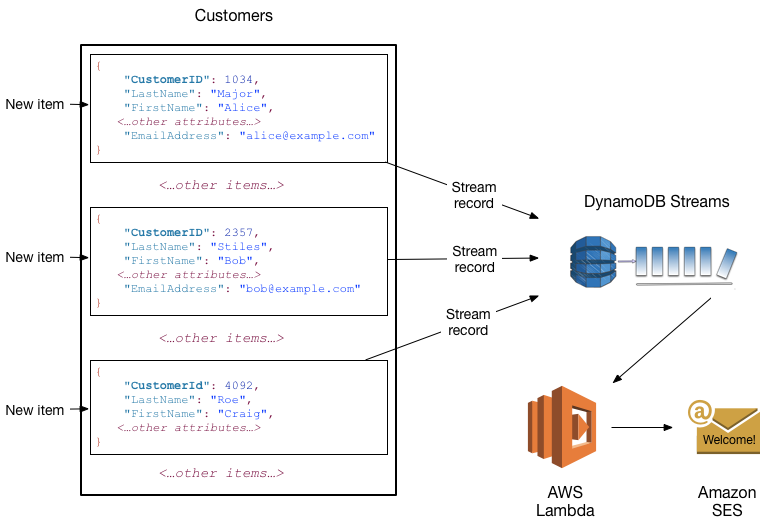
* + **Global table**
    - Có thể tạo replica table trên nhiều region khác
    - Bất kì thay đổi gì trên replica sẽ đồng bộ với table chính (mất 1 ít thời gian để đồng bộ, không support Strongly Consistent)
  + **Conditional writes** 
    - Check điều kiện khi 1 item update bởi nhiều request

|  |  |
| --- | --- |
|  |  |

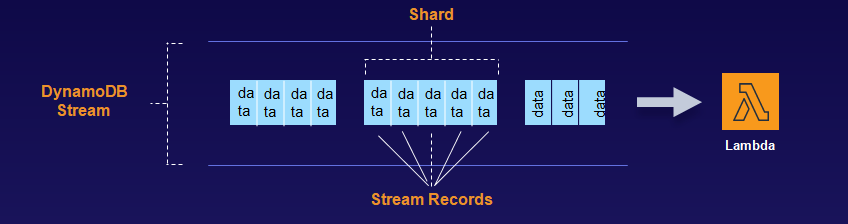
* + **Expiring Items By Time to Live (TTL)**
    - Có thể config attributetự hết hạn trong 1 thời gian
  + **Backup – Restore**
    - Backup table
    - Có 2 mode
      * On-demand backup
        + Tạo snapshot thủ công
        + Snapshot lưu tối đa 35 ngày (Giống RDS)
      * Enable Point-in-Time Recovery (PITR)
        + Cho phép restore về bất kì thời điểm nào (không cần tạo snapshot thủ công)
    - Khi restore có thể trên cross-region, setup encryption
  + **Export S3**
    - Có thể export table ra S3
  + **Encryption** (at rest)
    - Free
    - DynamoDB không encrypt từng item trong table 🡺 phải sử dụng client-side library để encrypt items trước khi lưu
    - Có 3 option client-side
      * Owned by Amazon DynamoDB
      * AWS managed CMK
      * Lưu trong account của mình và mình quản lý
  + **Monitoring**
    - CloudWatch
  + **DynamoDB Stream & Kinesis Data Streams**
    - Dành cho data stream
    - Item-level
    - Usecase
      * Các app có hàng nghìn lượt update/second
      * App chứng khoán
      * Ghi log các thay đổi real-time trong database (Insert, update, delete) trong 1 giờ
      * …
    - Có 2 loại
      * Kinesis Data Stream for DynamoDB
      * DynamoDB Stream



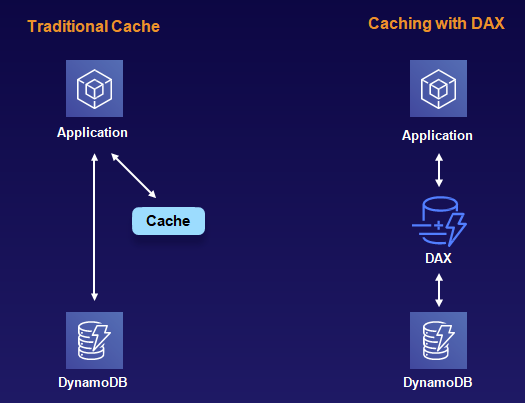
[*https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/streamsmain.html#streamsmain.choose*](https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/streamsmain.html#streamsmain.choose)



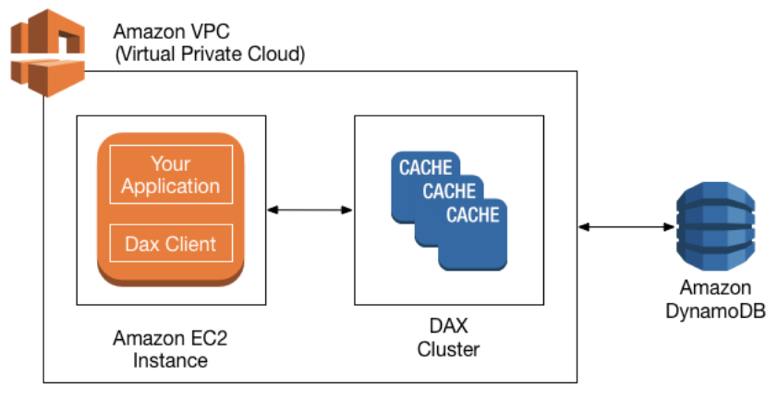
*stream record*

**

* + **DynamoDB Accelerator** - In-Memory Acceleration (DAX)
    - Là DynamoDB-compatible caching service
    - Lưu in-memory cache cho read giữa application & DynamoDB giúp tăng performance



* + - Các thành phần



* + - * DAX cluster
        + Bao gồm:

Các node

Network

Security

Maintenance

* + - * Node
        + Mỗi node là 1 EC2 Instance
        + Các node phải có type giống nhau
        + Thường có 3 node / 1 cluster, tối đa 11 node (cả 1 primary node)
        + Có 2 loại

Primary node

Có quyền write

Chỉ có 1 primary node

Read replicas

Có thể có nhiều Read replicas cho mục đích để cho Scalability & High availability

Khi primary node fails over thì sẽ tự assign 1 Read replicas lên là primary node

* + - * Network
        + Cluster sẽ run trong Subnet, VPC
        + Config

VPC, subnet

security group

* + - * Security

Config

* + - * + IAM role để cho phép cluster read/write DynamoDB
        + Encryption

Support cả

Encryption at rest (Server-side)

Encryption in transit (TLS - x509 certificate)

* + - * DAX client
        + Sau khi config xong sẽ deploy cluster thành DAX client trên EC2 instance
      * Query cache & Item cache
        + Item cache để lưu query
        + Query cache

Để lưu query khi sử dụng Query hoặc Scan

|  |  |
| --- | --- |
| *Item cache* | *Query cache* |

* **Config**
  + Setup

<https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/GettingStartedDynamoDB.html>

* **Reference**

<https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/Introduction.html>