Amazon DynamoDB

TCSS562: Group 7

Resham Ahluwalia
Savita Rana
SriHarshitha Somaraju
Priyanka Konduru

Presentation Overview

The key points of the presentation:

- Introduction and History of DynamoDb
- Motivation and Features
- Use cases
- Advantages and Disadvantages of using DynamoDb
- Usability
- Cost Analysis of using DynamoDB
- Comparison with other cloud provider
- Demo
- Conclusion

Amazon DynamoDb

- NoSQL database service that is offered by Amazon.com
- Amazon released DynamoDb on January 18, 2012.
- Amazon DynamoDB lets you create a database table that can store and retrieve any amount of data, and serve any level of request traffic.
- Supports both document and key-value store models.

Motivation

The reason for the introduction of NoSQL is because RDBMS cannot actually cope with some modern applications.This is because RDBMS has the following problems:

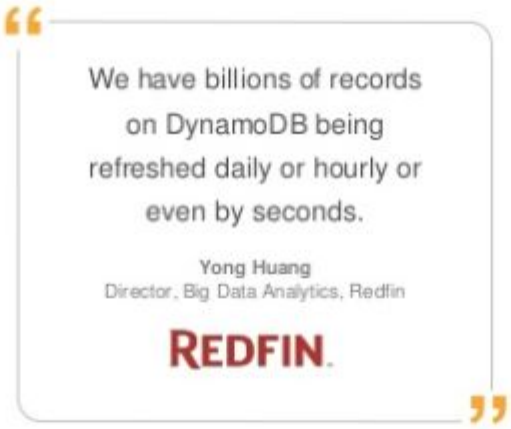
- Low throughput
- Only vertical scaling
- Object –Relational mapping is expensive
- The thinking "one size fits all" is flawed.
- ACID (Atomicity, Consistency, Isolation and Durability) is not always needed.

Features

- Automatic data replication
- Infinitely scalable read-write I/O running on IOPS-optimized solid state drives.
- A provisioned-throughput model
- Amazon DynamoDB Accelerator (DAX)
- Data backed up to S3
- Integrated with other AWS services
- Pay-per-use model
- Security and access control

Use Case - I

Redfin Manages Data on Hundreds of Millions of Properties Using Amazon DynamoDB

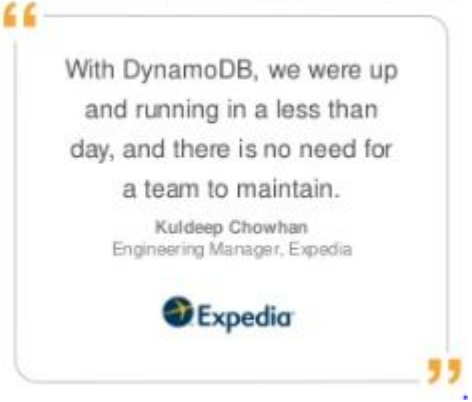


> We have billions of records on DynamoDB being refreshed daily or hourly or even by seconds.
>
> Yong Huang
> Director, Big Data Analytics, Redfin

REDFIN.

Use Case - II

Expedia’s real-time analytics applications uses DynamoDB



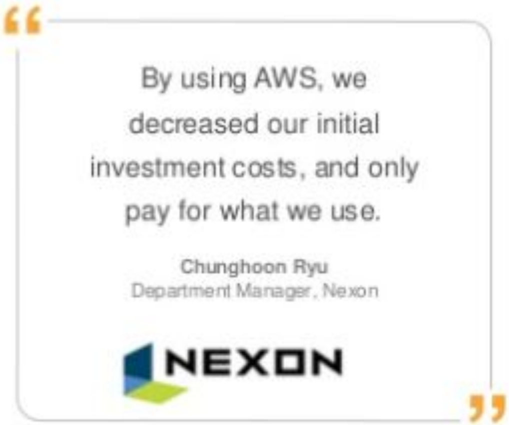
> With DynamoDB, we were up and running in a less than day, and there is no need for a team to maintain.

Kuldeep Chowhan
Engineering Manager, Expedia

Expedia

Use Case - III

Nexon uses DynamoDb as its primary game database for a new blockbuster mobile game, HIT



> By using AWS, we decreased our initial investment costs, and only pay for what we use.
>
> **Chunghoon Ryu**
> Department Manager, Nexon

Advantages

Fast, Consistent Performance

Highly Scalable

Fully Managed

Event Driven Programming

Fine-grained Access Control

Flexible

Disadvantages

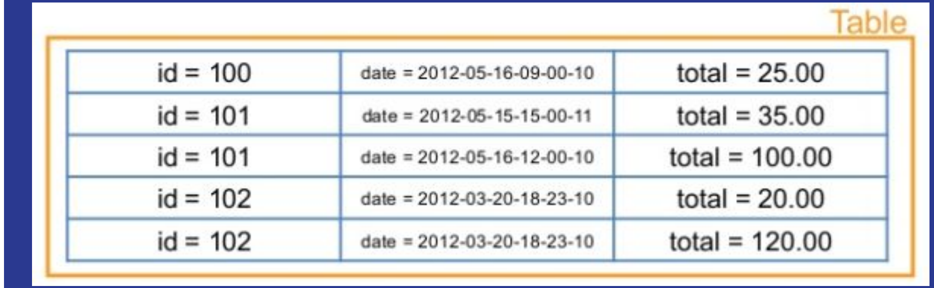
- 64 KB limit on row size
- Consistency comes with cost
- Attributes values cannot be null or empty
- Joins are impossible
- Indexing

Usability

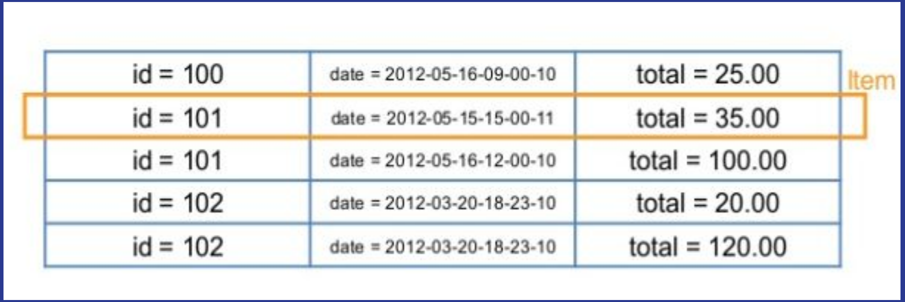
- Supports various platforms (.NET, PHP, Java, Python, Ruby, etc)
- The pricing is very simple
- The data model is very flexible, use of JSON
- Auto Scaling
- DAX, in-memory cache that can reduce DynamoDB response times from milliseconds to microseconds making its usability very easy
- Fully managed cloud database

DynamoDb Data Model

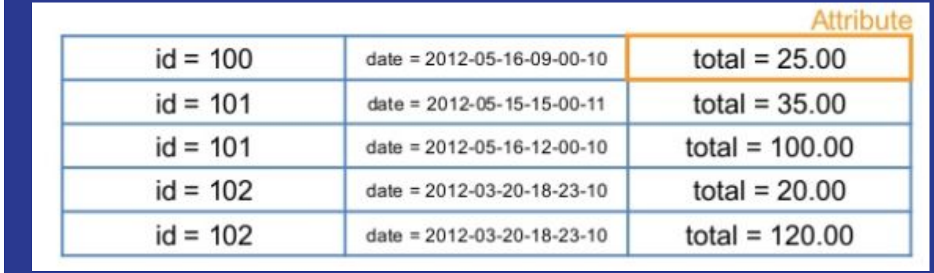
- DynamoDb is schemaless,except primary key
- Each item can have any number of attributes.
- An attribute is a name-value pair



id = 100	date = 2012-05-16-09-00-10	total = 25.00
id = 101	date = 2012-05-15-15-00-11	total = 35.00
id = 101	date = 2012-05-16-12-00-10	total = 100.00
id = 102	date = 2012-03-20-18-23-10	total = 20.00
id = 102	date = 2012-03-20-18-23-10	total = 120.00



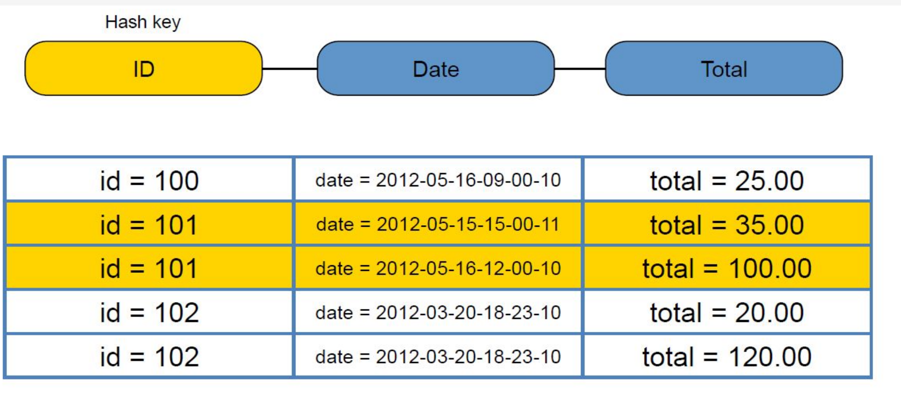
id = 100	date = 2012-05-16-09-00-10	total = 25.00
id = 101	date = 2012-05-15-15-00-11	total = 35.00
id = 101	date = 2012-05-16-12-00-10	total = 100.00
id = 102	date = 2012-03-20-18-23-10	total = 20.00
id = 102	date = 2012-03-20-18-23-10	total = 120.00



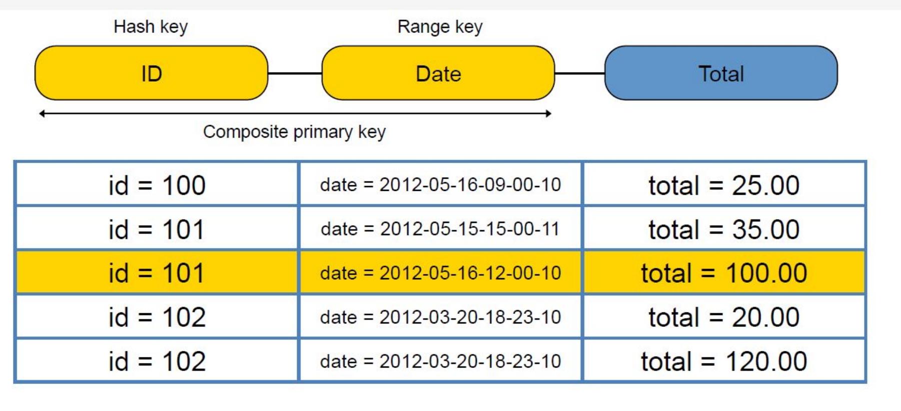
id = 100	date = 2012-05-16-09-00-10	total = 25.00
id = 101	date = 2012-05-15-15-00-11	total = 35.00
id = 101	date = 2012-05-16-12-00-10	total = 100.00
id = 102	date = 2012-03-20-18-23-10	total = 20.00
id = 102	date = 2012-03-20-18-23-10	total = 120.00

Data model concepts

- Primary keys
  - Hash type
  - Hash and Range type
- Secondary Indexes
  - Local Secondary
  - Global Secondary
- DynamoDb data types
  - Scalar Data types
  - Multivalued Data types



id = 100	date = 2012-05-16-09-00-10	total = 25.00
id = 101	date = 2012-05-15-15-00-11	total = 35.00
id = 101	date = 2012-05-16-12-00-10	total = 100.00
id = 102	date = 2012-03-20-18-23-10	total = 20.00
id = 102	date = 2012-03-20-18-23-10	total = 120.00



id = 100	date = 2012-05-16-09-00-10	total = 25.00
id = 101	date = 2012-05-15-15-00-11	total = 35.00
id = 101	date = 2012-05-16-12-00-10	total = 100.00
id = 102	date = 2012-03-20-18-23-10	total = 20.00
id = 102	date = 2012-03-20-18-23-10	total = 120.00

Cost Discussion

- Terms involved in DynamoDB pricing:
  - WCU: Number of item writes per second x item size in 1KB blocks
  - RCU: Number of item reads per second x item size in 4KB blocks
  - Target utilization= Real consumed capacity / total provisioned capacity

Pricing:

Resource Type	**Details**	**Monthly Price**
Provisioned Throughput (Write)	One write capacity unit (WCU) provides up to one write per second, enough for 2.5 million writes per month	As low as $0.47 per WCU
Provisioned Throughput (Read)	One read capacity unit (RCU) provides up to two reads per second, enough for 5.2 million reads per month	As low as $0.09 per RCU
Indexed Data Storage	DynamoDB charges an hourly rate per GB of disk space that your table consumes	As low as $0.25 per GB

Cost Discussion

AutoScaling Example:
Create a new table in US East (N.Virginia) region
Target utilization-70%
Minimum capacity units to 100 RCUs and 100 WCUs
Maximum capacity set to default limit of 40,000 RCUs and 40,000 WCUs.

During first hour: AutoScaling will not be triggered if actual utilization is with in target utilization
 Bill for the hour would be $0.065 for the 100 WCUs provisioned ($0.00065 * 100), and $0.013 for the 100 RCUs ($0.00013 * 100).

During second hour: If consumed utilization increased to 100WCU's and 100RCU's, resulting in provisioned capacity of 143 RCUs and 143 RCUs (% of 100 consumed ÷ 143 provisioned).
The bill for this second hour would be $0.0925 for 143 WCUs and $0.01859 for 143 RCUs.

Monthly bill for the table is the sum total of the hourly pricing of the capacity provisioned by Auto Scaling through the month.

Cost Example(US East Region):

Number of requests requests per second -100 (Writes)

Number of reads per second - 200 (This data is read twice)

Data Size -2Kb

Cost for S3(Monthly):

Total Number of Write Requests : 30*24*60*60*100= 259.2 Million Requests per month

Total Storage- 494.3GB

Cost for write requests - $1296 (Pricing: PUT, COPY, POST, or LIST Requests-$0.005 per 1,000 requests)

Cost for reads-$207.36 (Pricing:$0.0004 per 1,000 requests)

Cost for storage: $11 (0.023 per GB)

Cost for data returned: $0.346 (Data Returned by S3 Select-$0.0007 per GB)

Total Cost: $1500 Approximately

Cost Using DynamoDB

Number of WCUs- 200(Since 2KB)

Monthly Cost- 200WCUs * $0.47per WCU= $94

Number of RCUs- 200 (Since data is read twice)

Monthly Cost- 200RCUs * .09per RCU= $18

Cost for storage-494.3GB * .25per GB= $123.575

Total Cost- $235(**6X times cheaper than S3**)

That’s not all.

If data size per request is 50KB instead of 2KB.

Cost for S3 would be approximately $3000 per month whereas DynamoDB would cost approximately 6500 (**Almost twice expensive as S3**)

Alternatives

Features	**DynamoDB**	**Google Big Table**
Architecture	Decentralised	Centralised
DataModel	Key Value	Sorted Map
API	Get, Put	Get, Put, Scan, Delete
Security	No	Access Control
Partitioning	Consistent Hashing	Key Range Based
Replication	Successor Nodes in ring	Chunk Servers in GFS
Storage	Plugin	SS Table in GFS
Membership and	Gossip Based Protocol	Handshakes

Conclusions

- Fast, Flexible non relational database
- Reliable performance and Automated scaling
- Great fit for mobile, web, gaming, ad tech, IoT, and many other applications.
- Cost Effective in some use cases.

DEMO

References

- https://www.uniassignment.com/essay-samples/information-technology/review-of-the-motivation-for-nosql-information-technology-essay.php
- https://cloudacademy.com/blog/amazon-dynamodb-ten-things/
- https://aws.amazon.com/dynamodb/
- https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/GettingStarted.Java.html
- https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/JavaDocumentAPICRUDExample.html

Questions???