RDS
(Relational Database Service)

Managed Services

Managed Services

Scaling, fault tolerance and availability managed by AWS

e.g.-

RDS

Amazon S3

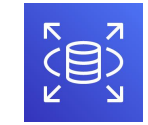
DynamoDB

Unmanaged Services

Scaling, fault tolerance and availability managed by you

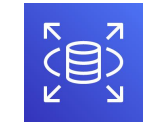
e.g.-

Amazon EC2



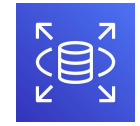
Challenges of Relational Databases

- Server maintenance
- Software installation
- Patching of Operating System
- High Availability
- Data Security
- Licenses



Definition

RDS is a managed service that sets up and operates a relational database in the AWS Cloud.



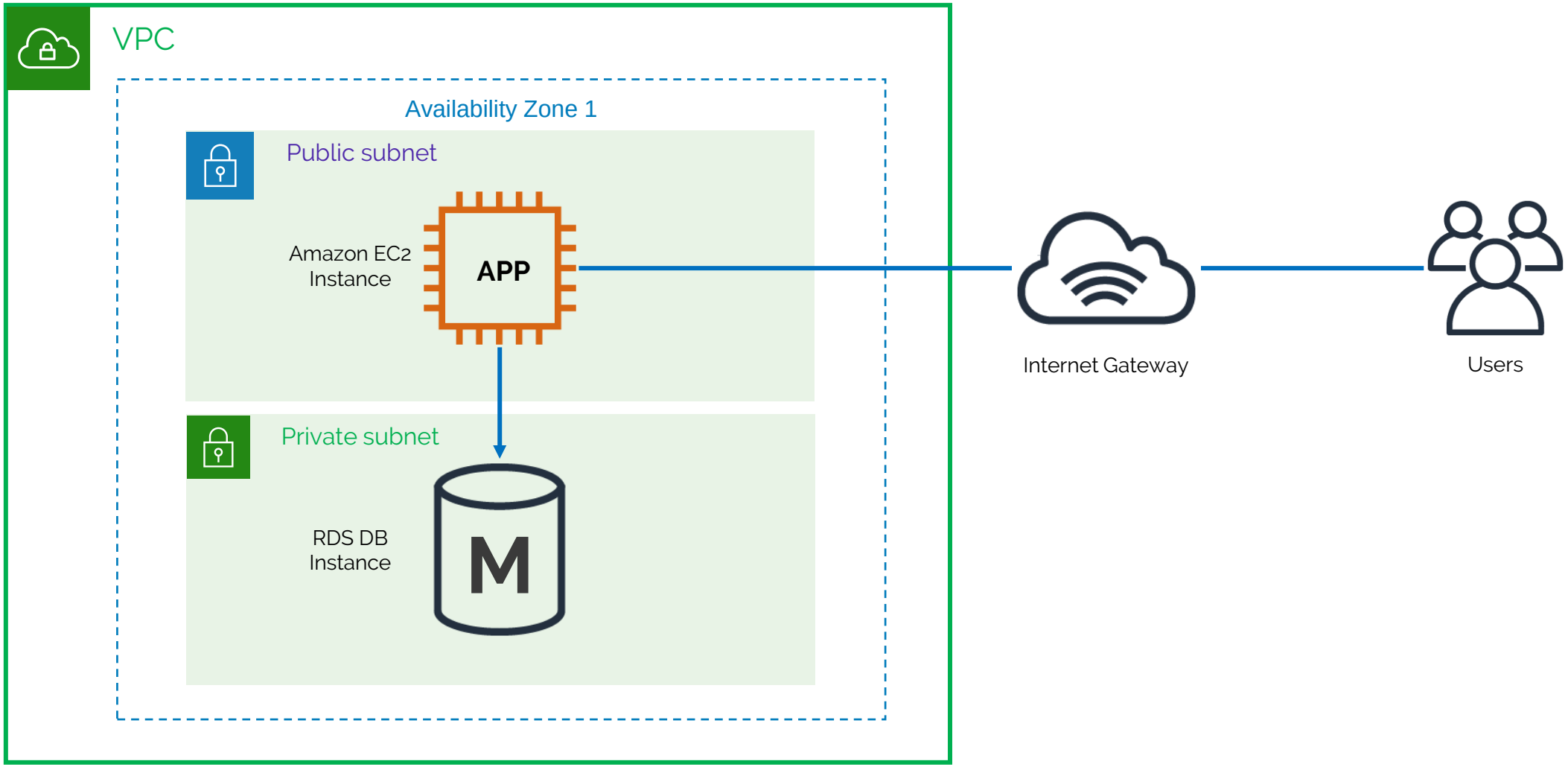
RDS Instance Types

Model	Core Count	vCPU*	CPU Credits/hour	Mem (GiB)	Network Performance (Gbps)
db.t3.micro	1	2	12	1	Up to 5
db.t3.small	1	2	24	2	Up to 5
db.t3.medium	1	2	24	4	Up to 5
db.t3.large	1	2	36	8	Up to 5
db.t3.xlarge	2	4	96	16	Up to 5
db.t3.2xlarge	4	8	192	32	Up to 5

Database Engines

Amazon Aurora

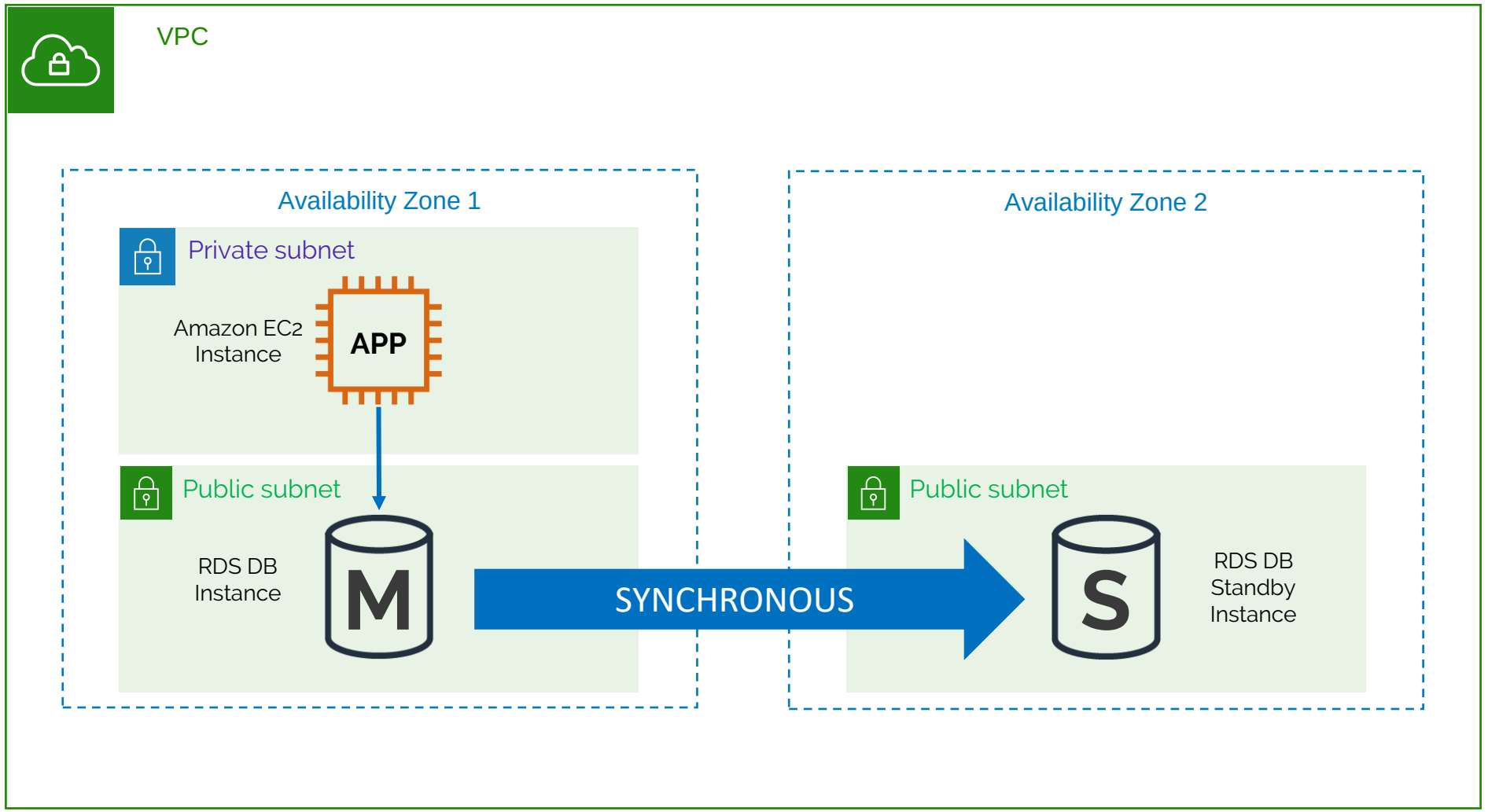
RDS in VPC


VPC
Availability Zone 1
Public subnet
Amazon EC2
Instance
APP
Private subnet
RDS DB
Instance
M
Internet Gateway
Users

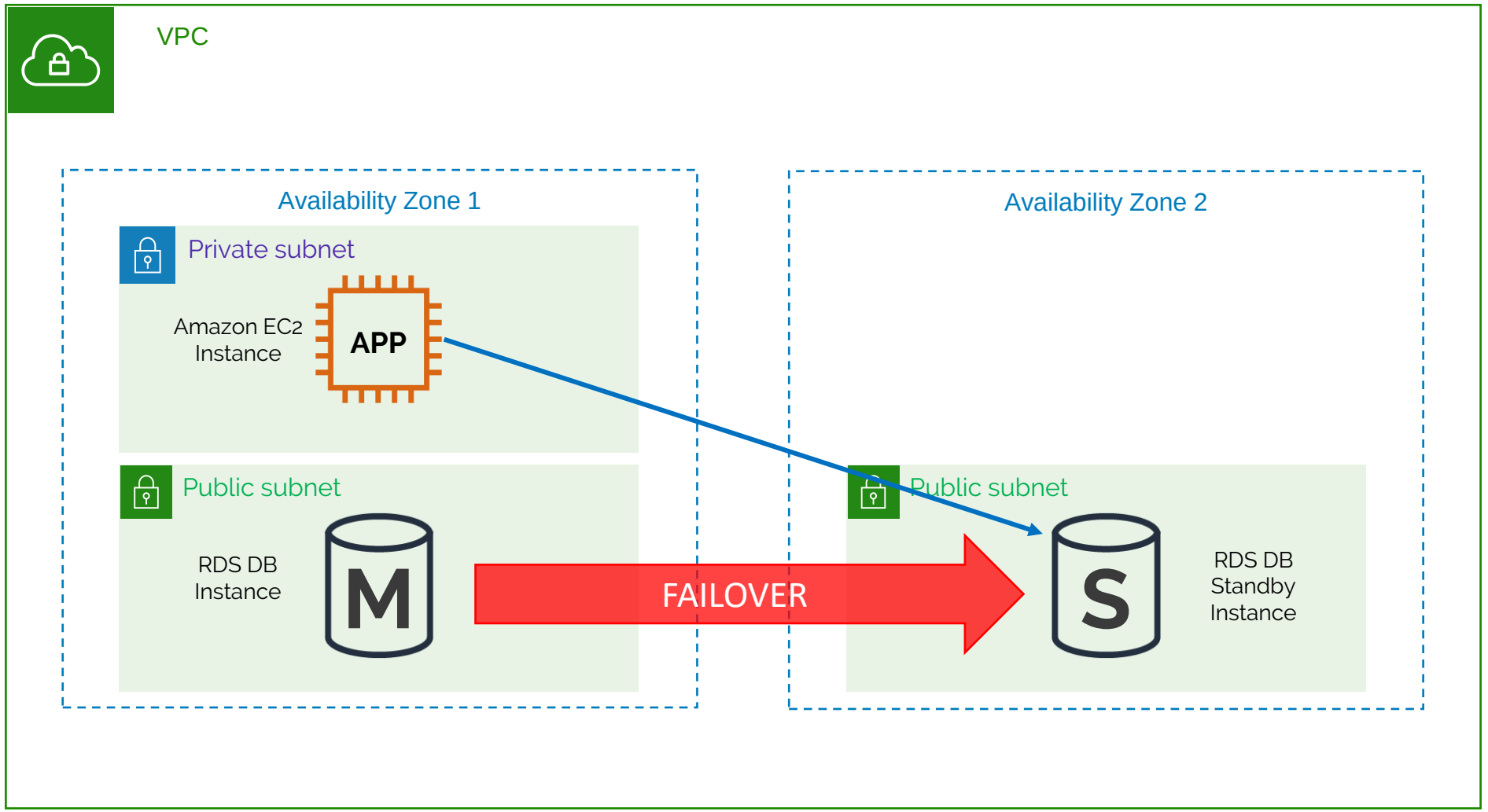
Backups

- Automated backups enabled by default; Retain transaction and database Logs
- Backup period up to 35 days
- Configurable Backup Window
- Manual Snapshots

Multi-AZ failover



Multi-AZ failover



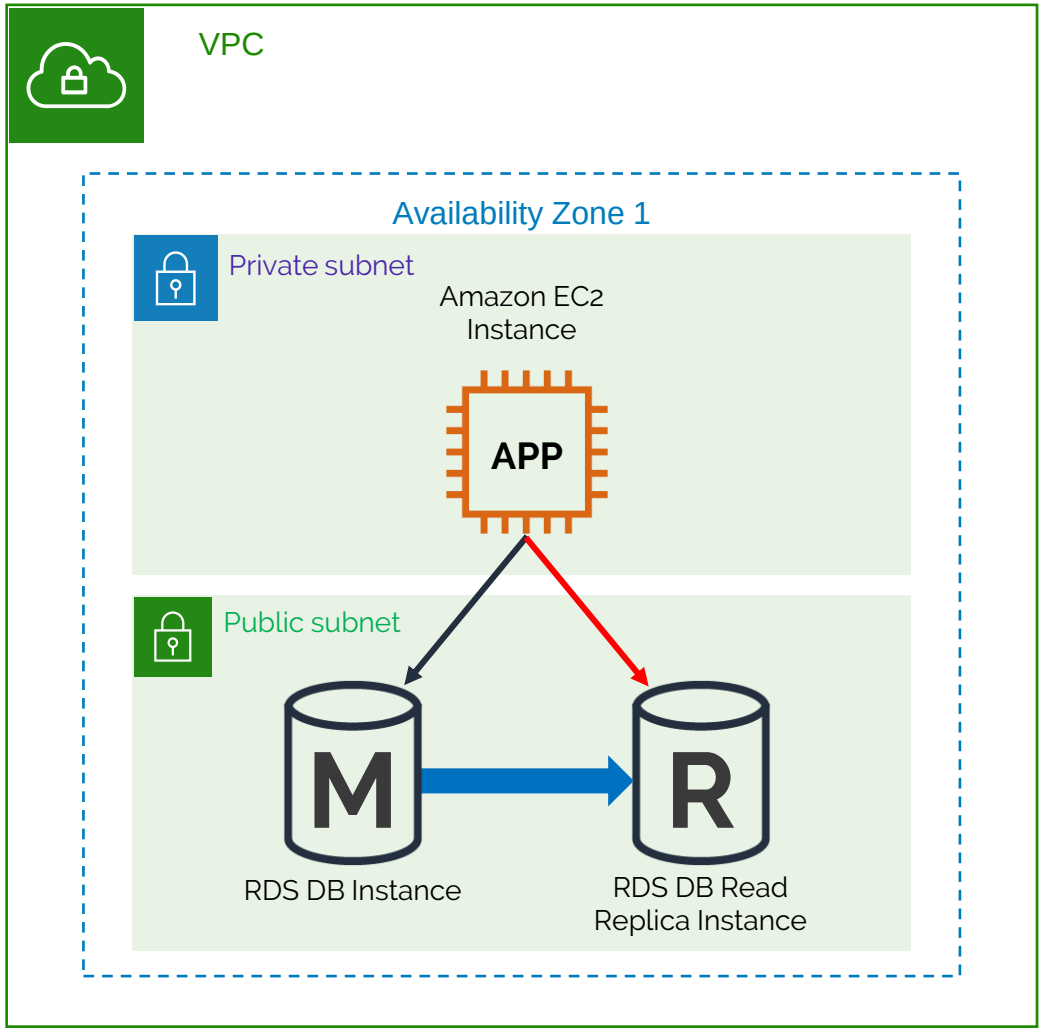
Multi-AZ

Database Failure Reasons:

- Host Failure
- Network Failure
- AZ failure
- Patching Maintenance

Read Replicas

- Features
  - Asynchronous replication
  - Promote to master if needed
- Functionality
  - Read-heavy database workloads
  - Offload read queries



Read Replica Features

- Can be created in a different Region
- Different storage type
- Up to 5 read replicas for MySQL, PostgreSQL and MariaDB
- Read Replicas can be Multi AZ
- Replication is Asynchronous only; Not Synchronous
- Create Aurora Read Replicas and Promote to Master

Use Cases of Read Replicas

- If source DB is not able to take I/O requests
- Business reporting scenarios
- Direct excess traffic for read heavy database workload
- Disaster recovery requirements
- Reduce latency for end users

Multi AZ vs Read Replica

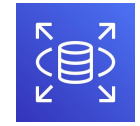
Multi-AZ Deployments	Read Replicas
Synchronous replication – highly durable	Asynchronous replication – highly scalable
Only database engine on primary instance is active	All read replicas are accessible and can be used for read scaling
Automated backups are taken from standby	No backups configured by default
Always span two Availability Zones within a single Region	Can be within an Availability Zone, Cross-AZ, or Cross-Region
Database engine version upgrades happen on primary	Database engine version upgrade is independent from source instance
Automatic failover to standby when a problem is detected	Can be manually promoted to a standalone database instance

When not to use RDS

- Massive read/write rates
- Division into Shards
- Customization of RDBMS
- Simple GET/PUT requests

Pricing

- Number of Hours that Database runs
- Physical capacity of Database
    - Database Engine
    - Instance Type
    - Storage Class
- On Demand and Reserved Pricing Model
- Number of Instances



NoSQL Basics and What is DynamoDB?

NoSQL and DynamoDB

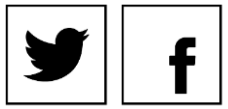
NoSQL

Gaming

IoT

Web

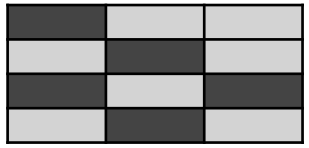
Mobile



Social

Document Database

Enterprise



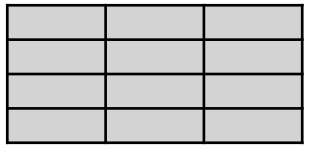
Column Family Store

SQL

Mobile

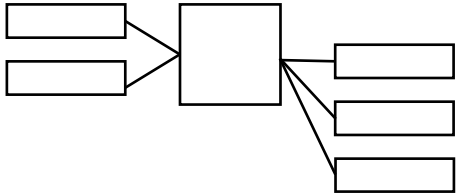
Web

Data-Mart



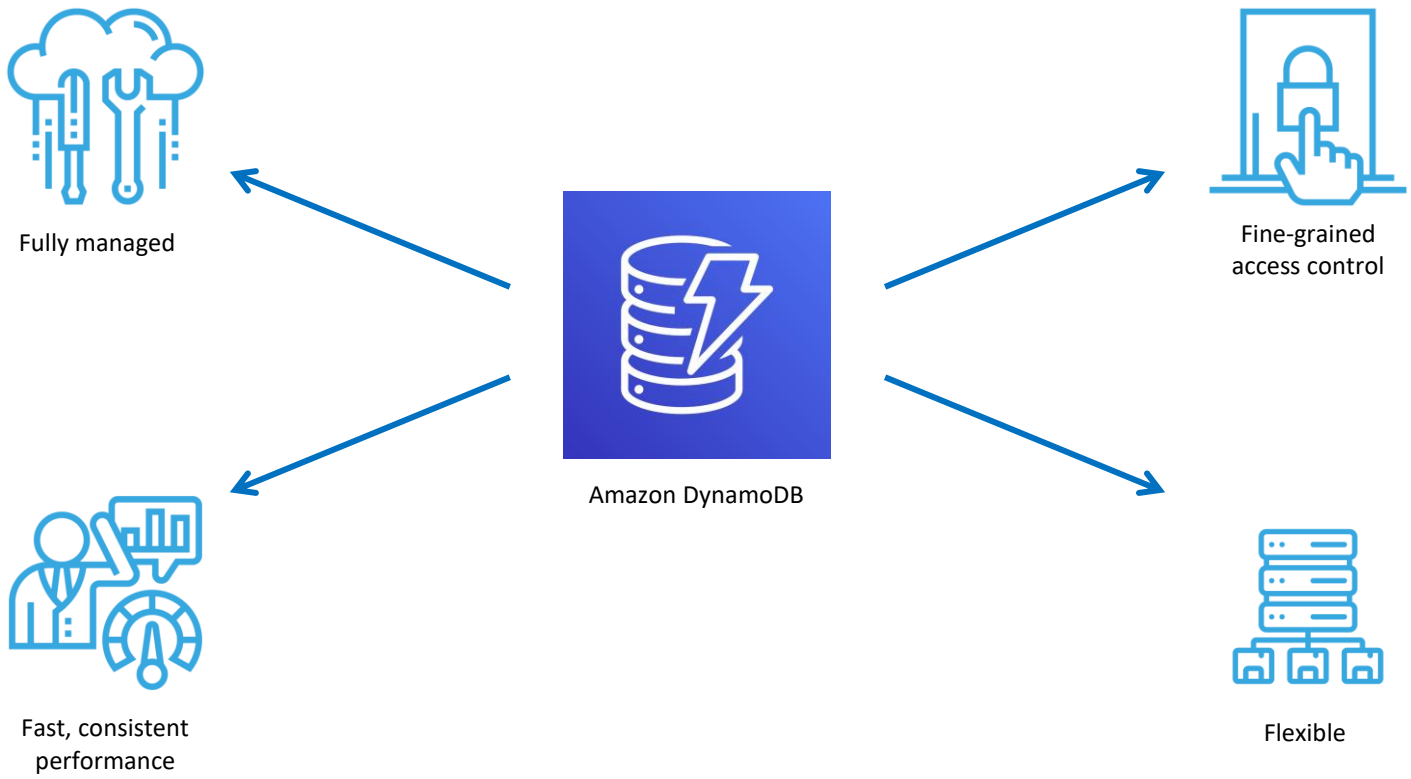
Relational table storage

Enterprise



Relationships use joins

Amazon DynamoDB

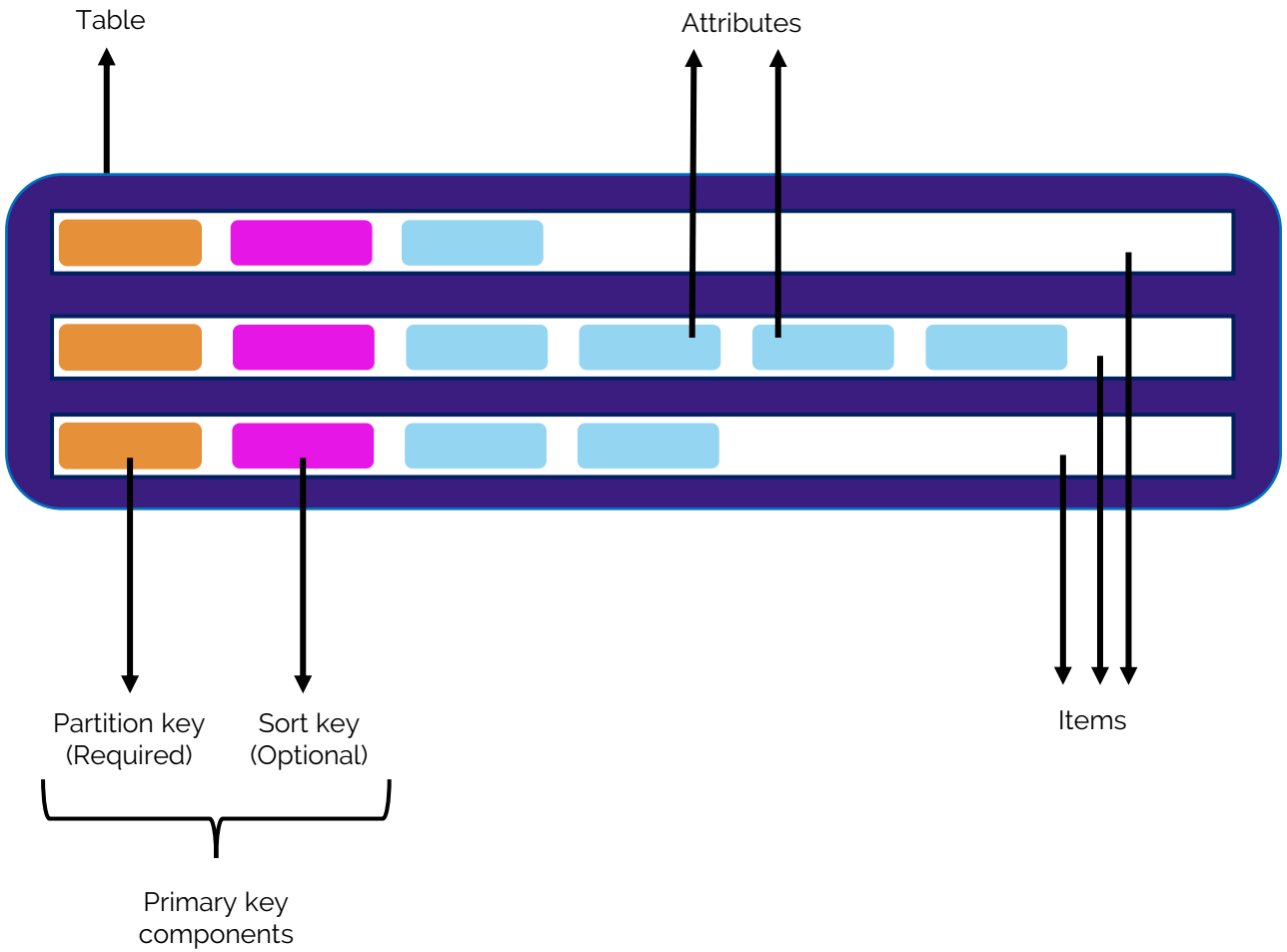


DynamoDB Use Cases

Amazon DynamoDB

- Microservices data store
- Serverless web applications
- Adtech
- Internet of Things
- Mobile backends
- Gaming

Components of DynamoDB



Types of Primary Keys

Simple Primary Key
(partition key only)

Artist	Album Title	Year
AR Rahman	Tamasha	2015
Kishore Kumar	Aaradhana	1969

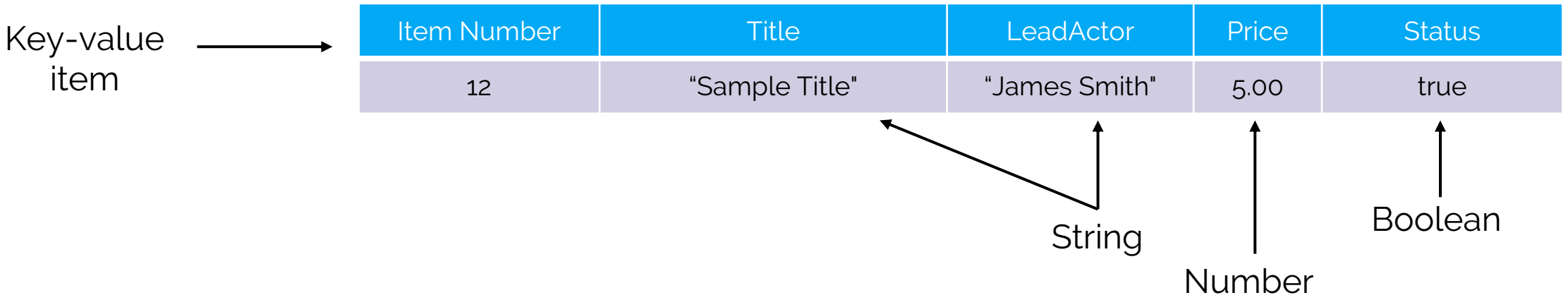
Album Table

Composite Primary Key
(partition key and sort key)

Artist	Song Title (Sort Key)	Year
AR Rahman	Tamasha	2015
AR Rahman	Agar Tum Saath ho	2015
Kishore Kumar	Roop Tera Mastana	1969

Artist Song Table

Items and Attribute Types



Document Format

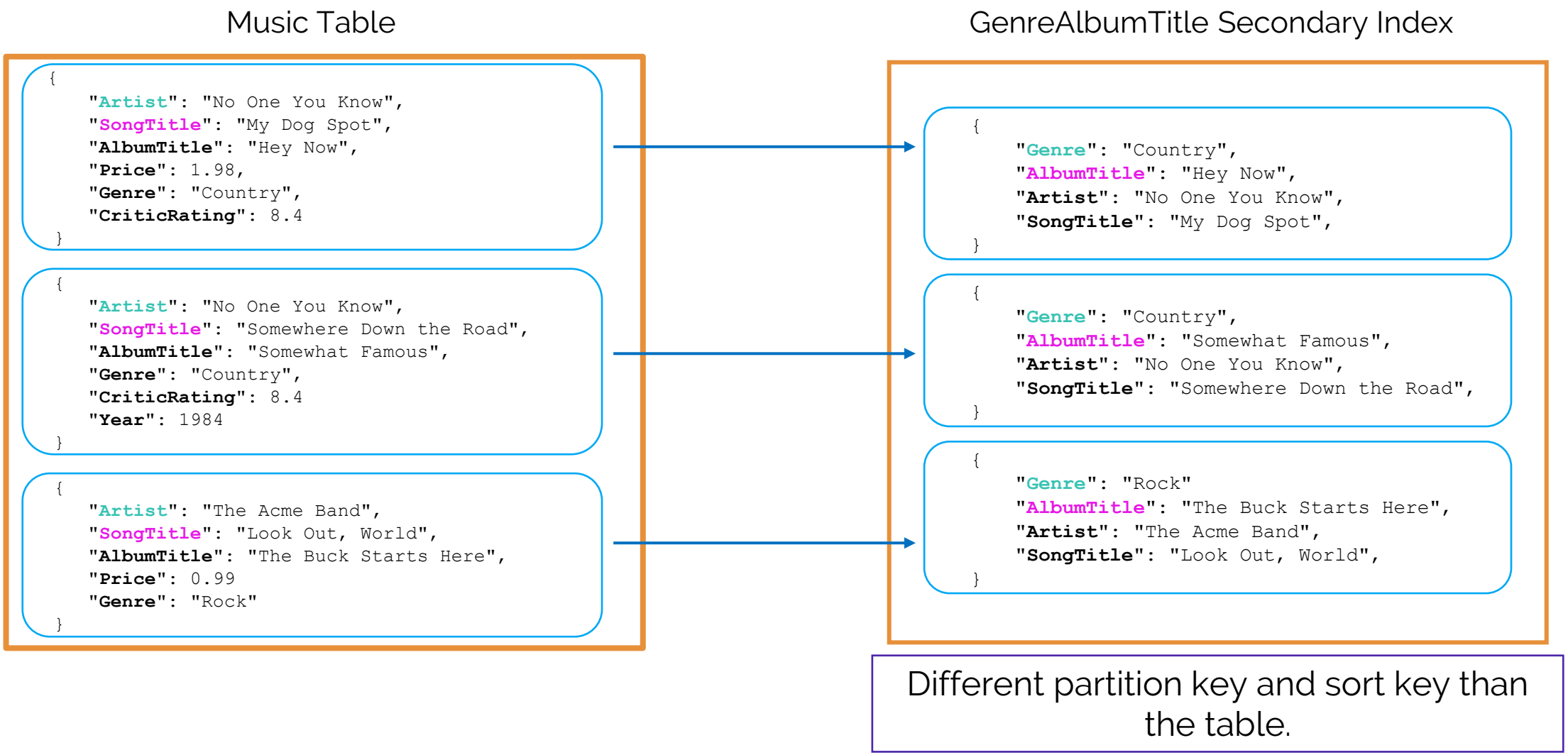
Document item →

```
{
  "year" : 2013,
  "title" : "Turn It Down, Or Else!",
  "info" : {
    "directors" : [
      "Alice Smith", "Bob Jones" ],
"image_url" : "http://ia.media-
imdb.com/images/N/O9ERWAU7FS797AJ7LU8H
N09AMUP908RLlo5JF90EWR7LJKQ7@@._V1_SX4
00_.jpg",
    "plot" : "A rock band plays their
music at high volumes, annoying the
neighbors."
}
}
```

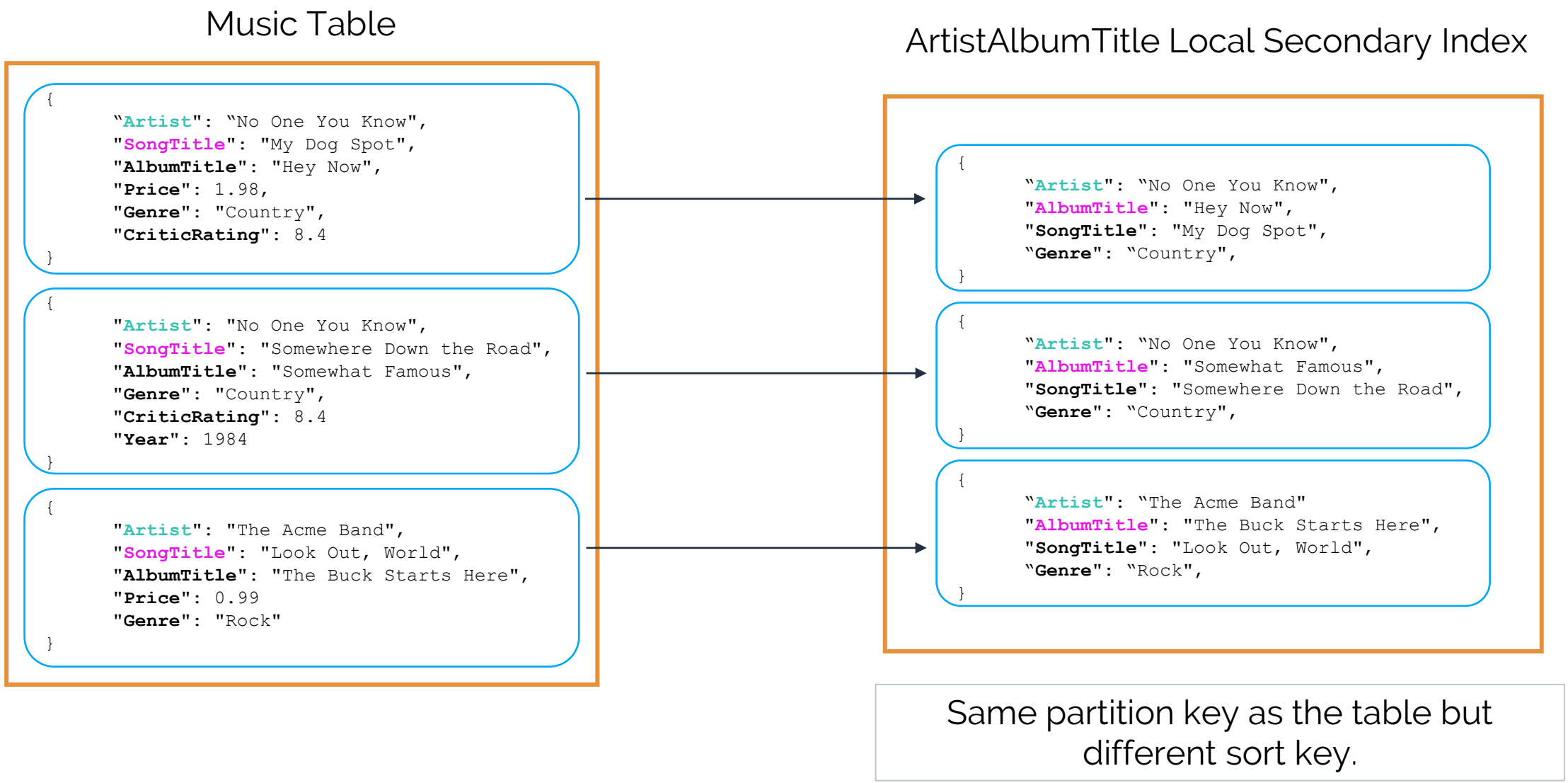
← Map

← List

Global secondary index example



Local secondary index example



Same partition key as the table but different sort key.

Consistency

- DynamoDB maintains multiple copies of data for durability.
- Consistency describes when changes are visible to everyone
- Copies usually consistent within a second after a write operation.
- Models define how quick you want changes to be updated

Consistency Models

- Eventually consistent
  - Read might return slightly stale data If read is done immediately after write
  - Use case: blog posts
  - Low latency
- Strongly consistent
  - Read returns most up-to-date data.
  - Use cases: score boards, booking systems, financial algorithms
  - High latency

Read/Write Consistency

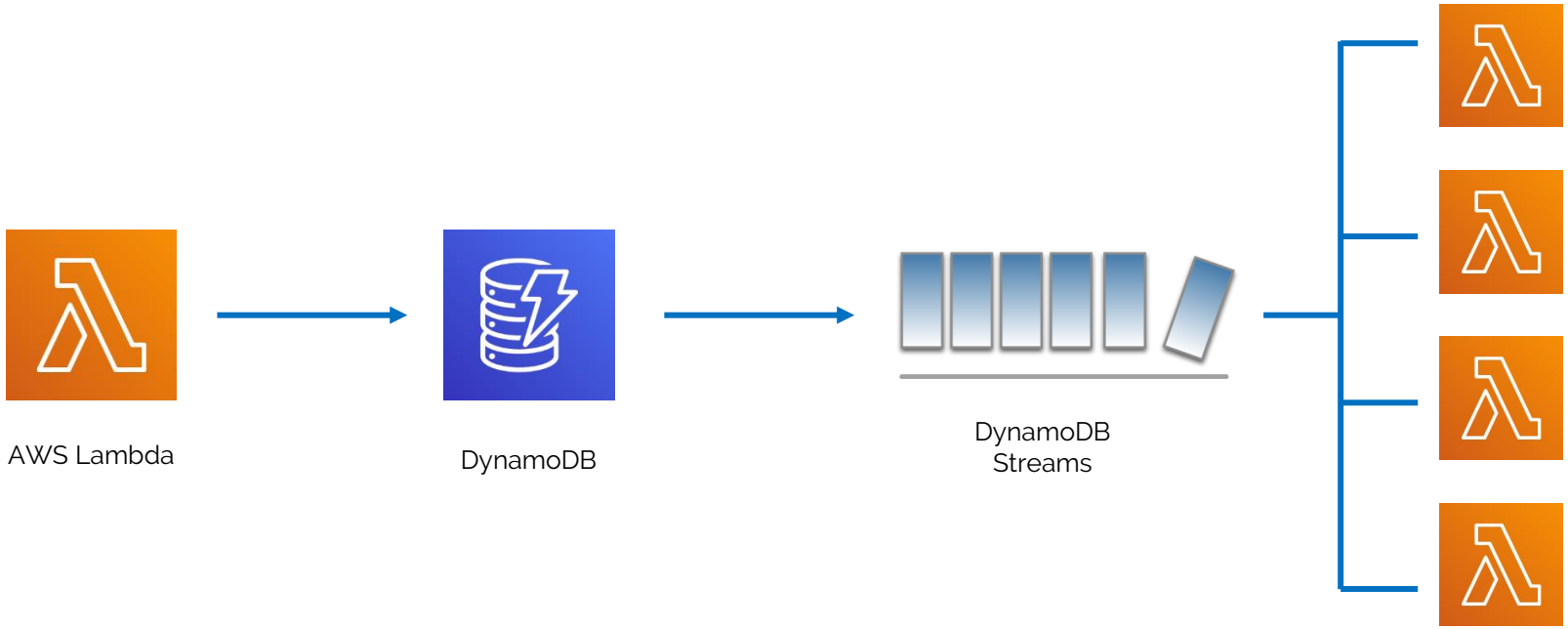
- Throughput = maximum amount of capacity that an application can consume from a table or index
- Read capacity unit (RCU)
  - Number of strongly consistent reads per second of items that are up to 4 KB in size
  - Eventually consistent reads use half of the provisioned read capacity
- Write capacity unit (WCU)
  - Number of 1 KB writes per second

DynamoDB Streams

- Capture and process changes to DynamoDB items using DynamoDB Streams
- Stores information in logs for up to 24 hours
- Logs can be processed by other applications in real time

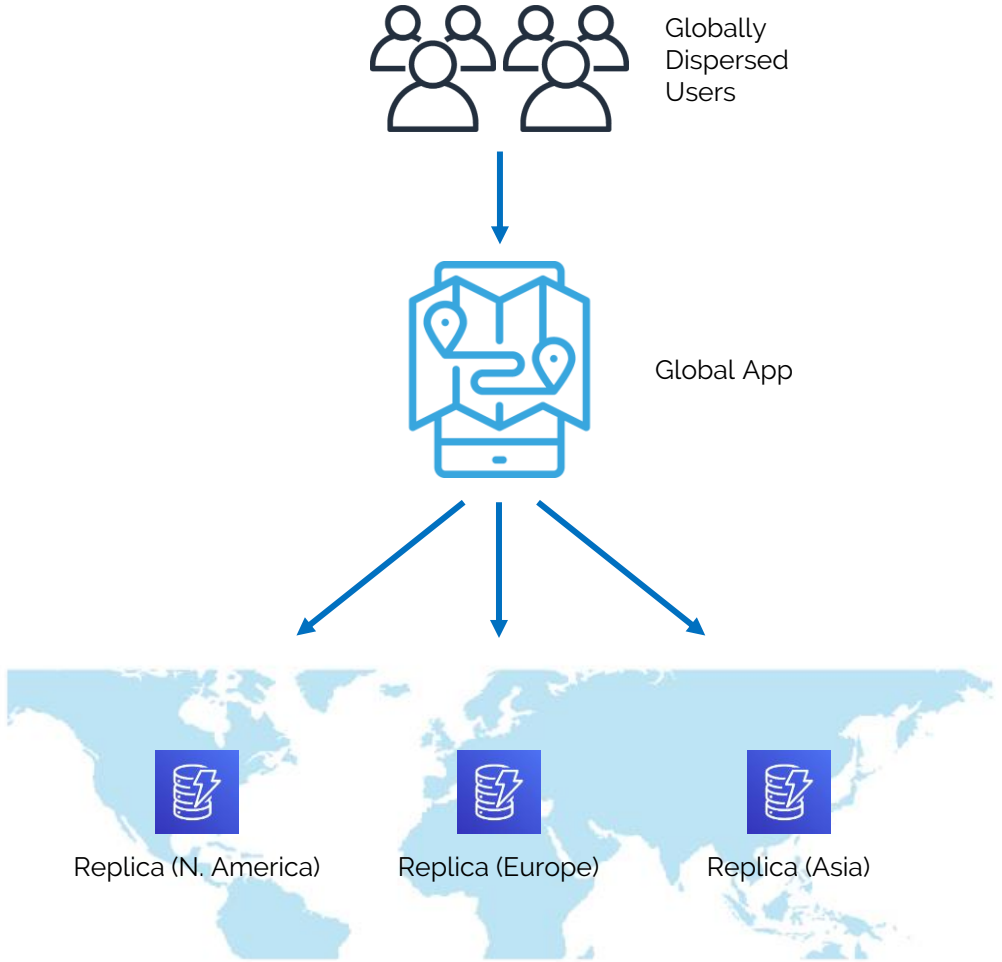
Real life Use case

User adding data to DynamoDB processed by Lambda through Streams



DynamoDB Global Tables

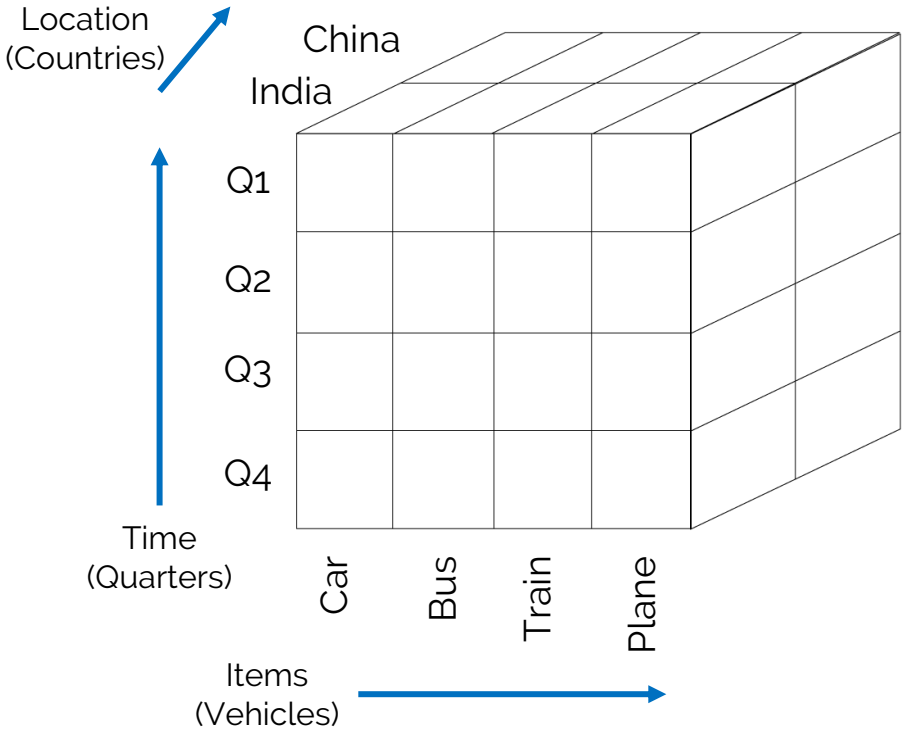
- Master Table to sync data across multiple regions
- Identical tables created in multiple regions
- Specify the AWS Regions where you want table to be replicated



Amazon Redshift

Online Analytical Processing

- Category of software used for analysing information from multiple databases at one point of time.
- Extract and view business data from different points.
- Store the huge amount of data for processing.



What exactly is Redshift?

- Fully managed
- Petabyte scale Data Warehouse service
- Columnar Processing
- Massive Parallel Processing

Redshift Components

- Compute Node with dedicated CPU, memory and storage to execute queries
- Leader Node

    Manages Communication

    Develops Execution Plans

    Assigns tasks to each compute node
- Compute node executes compiled code and send intermediate results back to leader node for final aggregation.

Use Cases

- Data Warehousing
- Log Processing
- Business Applications
- Data Reporting
- Predictive Models

Redshift pricing

- Pay for what you use
- On-Demand or Reserved
- Redshift Spectrum to run queries

Features

- Database to load and run queries
- Cluster groups in VPC
- Node types supported
    - Dense storage – Large data, HDD
    - Dense compute – Performance intensive, SSD
- Parameter Groups
- Modify, Delete, Reboot or Resize

Features (Continued)

- Standard SQL
    - Online Editor
    - Clients through ODBC and JDBC
- Accessible from Console, SDK, CLI and Redshift API
- Backups & Cross Region Snapshots
- Built in Encryption