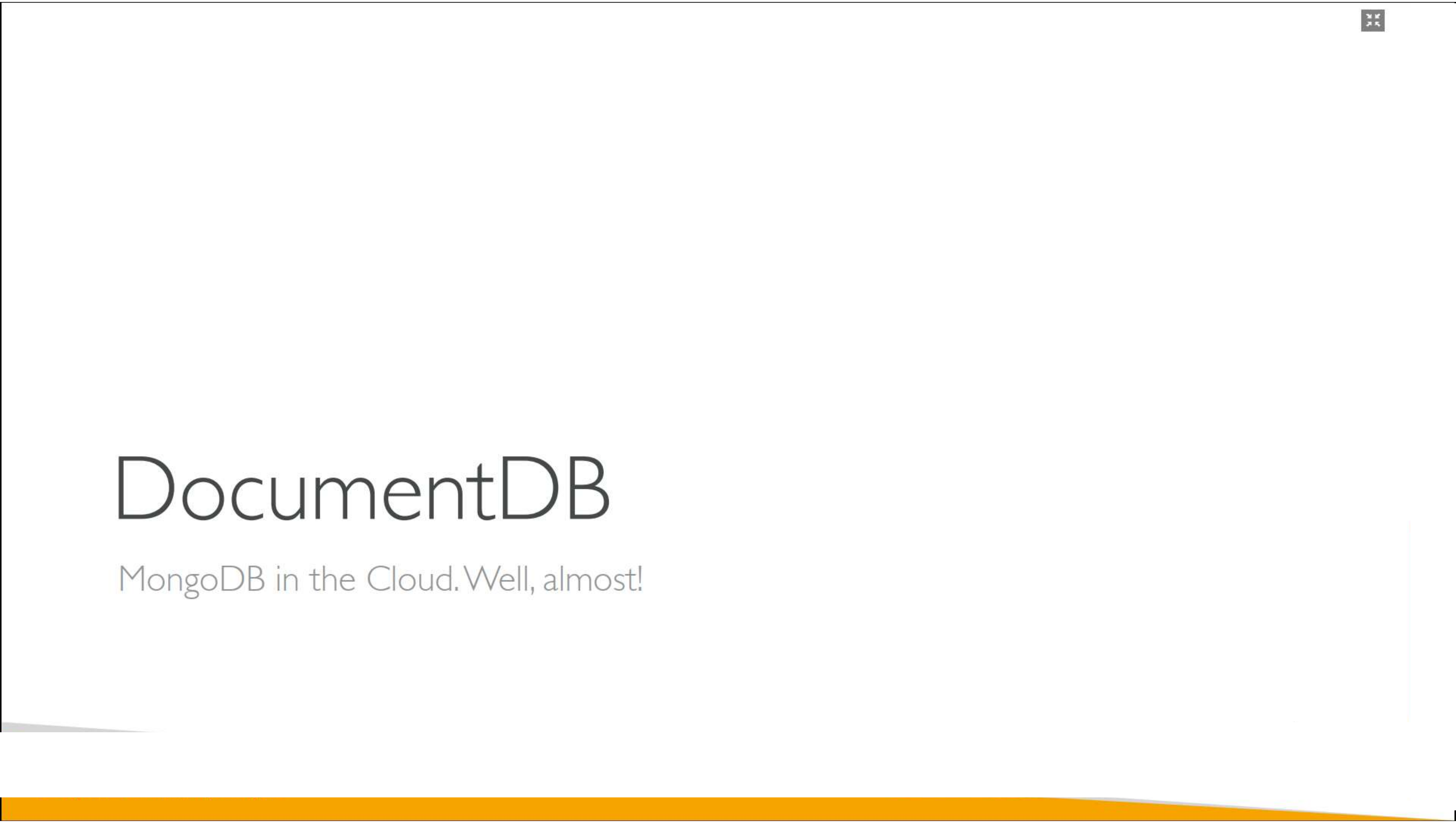
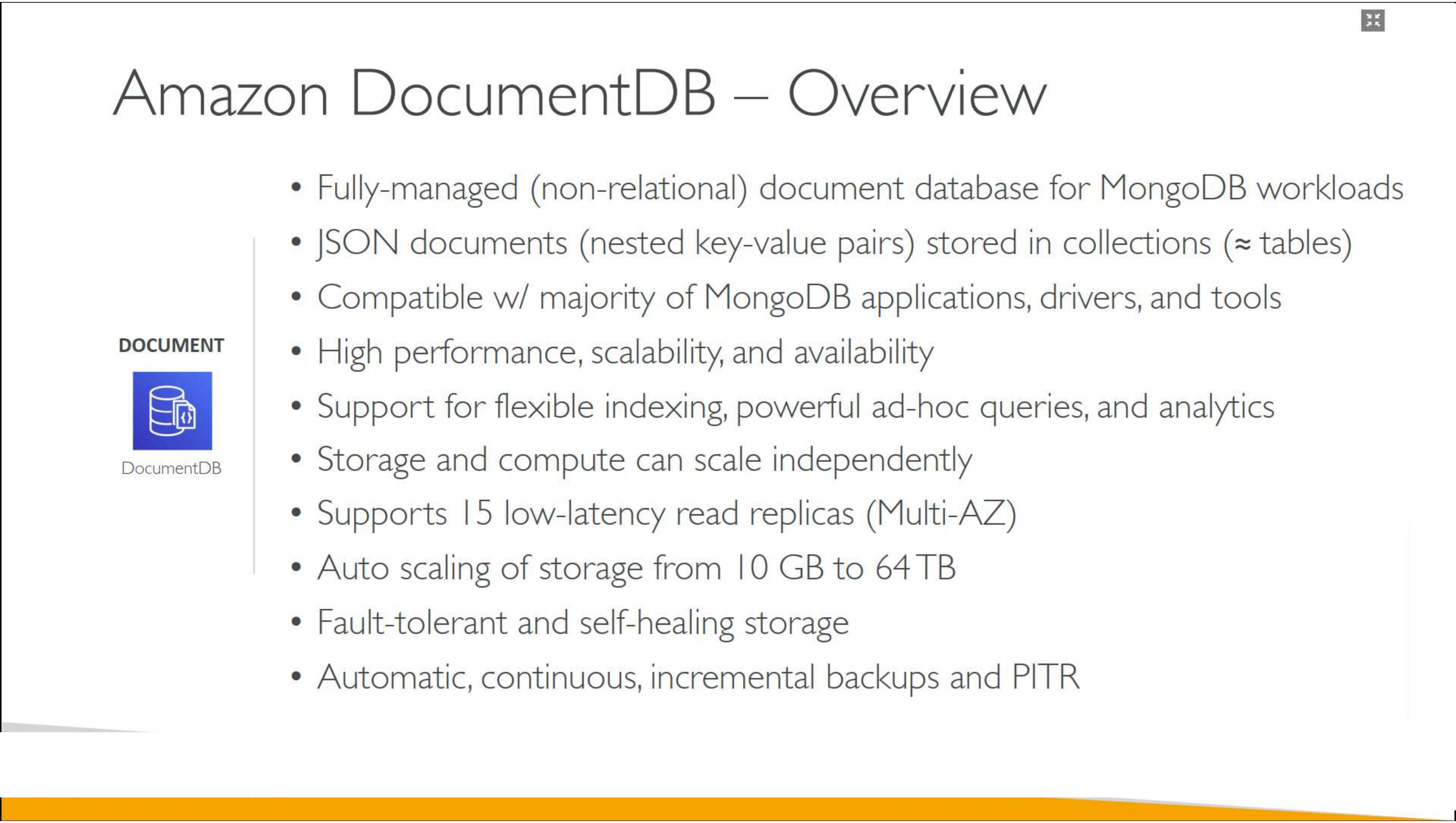


DocumentDB

MongoDB in the Cloud. Well, almost!



Amazon DocumentDB – Overview

DOCUMENT

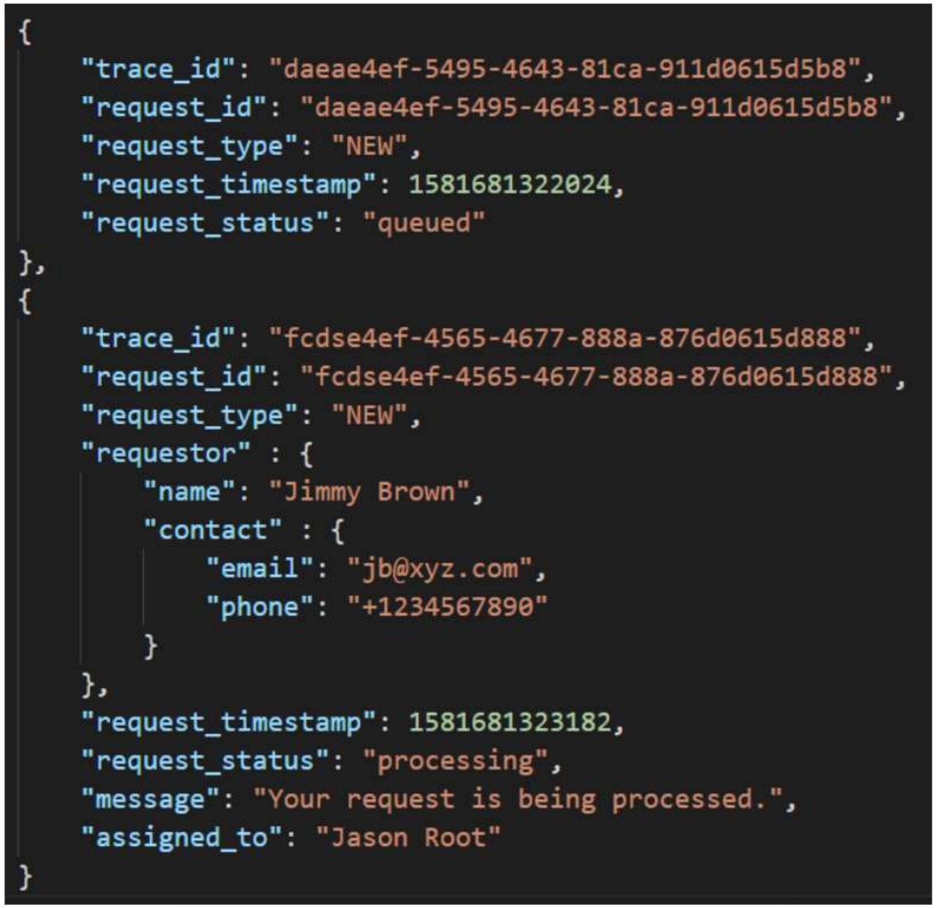
DocumentDB

- Fully-managed (non-relational) document database for MongoDB workloads
- JSON documents (nested key-value pairs) stored in collections (≈ tables)
- Compatible w/ majority of MongoDB applications, drivers, and tools
- High performance, scalability, and availability
- Support for flexible indexing, powerful ad-hoc queries, and analytics
- Storage and compute can scale independently
- Supports 15 low-latency read replicas (Multi-AZ)
- Auto scaling of storage from 10 GB to 64 TB
- Fault-tolerant and self-healing storage
- Automatic, continuous, incremental backups and PITR

Document Database

- Stores JSON documents (semi-structured data)
- Key-value pairs can be nested

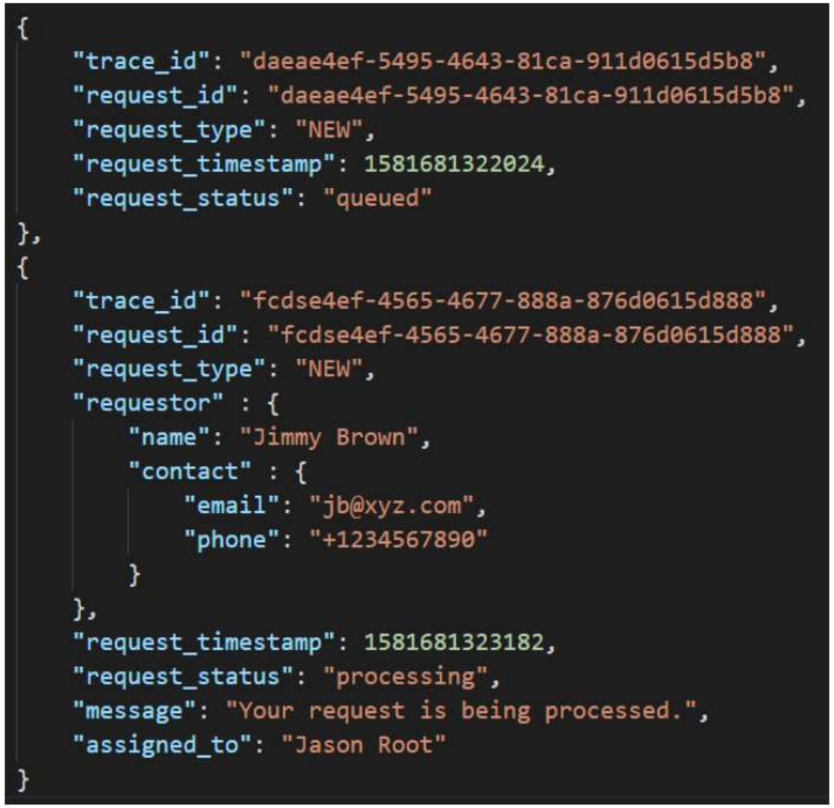
Relational DB (SQL)	DocumentDB (MongoDB)
Table	Collection
Rows	Documents
Columns	Fields
Primary key	Object ID
Nested table / object	Embedded documents
Index / view / array	Index / view / array



```
{
    "trace_id": "daeae4ef-5495-4643-81ca-911d0615d5b8",
    "request_id": "daeae4ef-5495-4643-81ca-911d0615d5b8",
    "request_type": "NEW",
    "request_timestamp": 1581681322024,
    "request_status": "queued"
},
{
    "trace_id": "fcdse4ef-4565-4677-888a-876d0615d888",
    "request_id": "fcdse4ef-4565-4677-888a-876d0615d888",
    "request_type": "NEW",
    "requestor" : {
        "name": "Jimmy Brown",
        "contact" : {
            "email": "jb@xyz.com",
            "phone": "+1234567890"
        }
    },
    "request_timestamp": 1581681323182,
    "request_status": "processing",
    "message": "Your request is being processed.",
    "assigned_to": "Jason Root"
}
```

Why document database?

- JSON is the de-facto format for data exchange
- DocumentDB makes it easy to insert, query, index, and perform aggregations over JSON data
- Store JSON output from APIs straight into DB and start analysing it
- flexible document model, data types, and indexing
- Add / remove indexes easily
- Run ad hoc queries for operational and analytics workloads
- for known access patterns – use DynamoDB instead

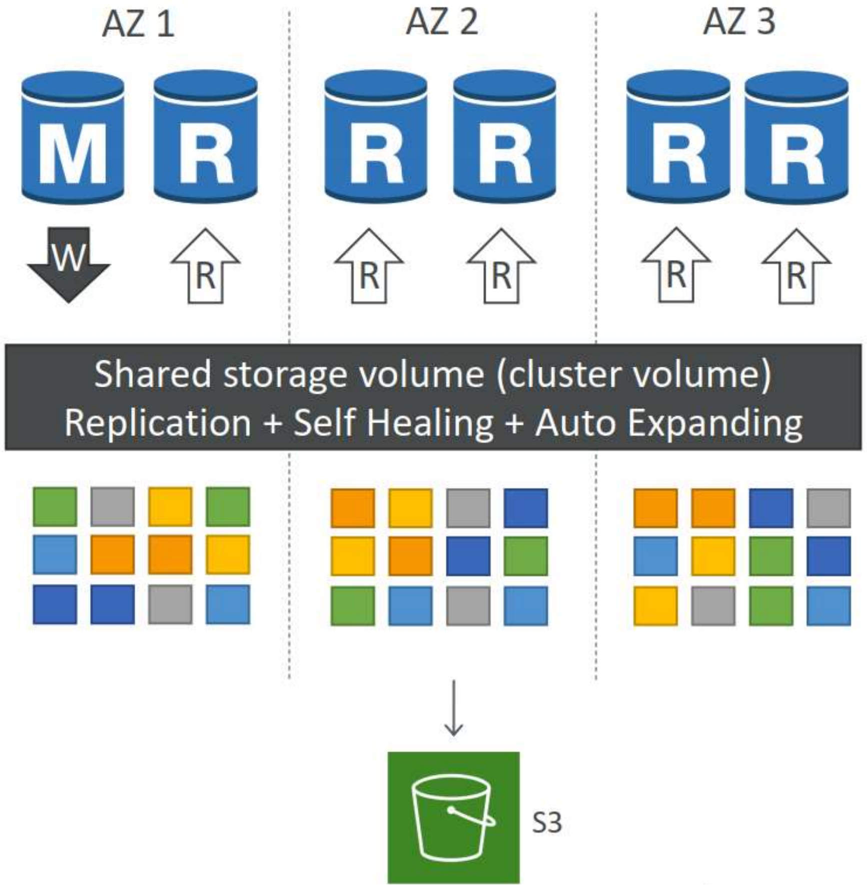


```
{
    "trace_id": "daeae4ef-5495-4643-81ca-911d0615d5b8",
    "request_id": "daeae4ef-5495-4643-81ca-911d0615d5b8",
    "request_type": "NEW",
    "request_timestamp": 1581681322024,
    "request_status": "queued"
},
{
    "trace_id": "fcdse4ef-4565-4677-888a-876d0615d888",
    "request_id": "fcdse4ef-4565-4677-888a-876d0615d888",
    "request_type": "NEW",
    "requestor" : {
        "name": "Jimmy Brown",
        "contact" : {
            "email": "jb@xyz.com",
            "phone": "+1234567890"
        }
    },
    "request_timestamp": 1581681323182,
    "request_status": "processing",
    "message": "Your request is being processed.",
    "assigned_to": "Jason Root"
}
```

DocumentDB Architecture

- 6 copies of your data across 3 AZ (distributed design)
  - Lock-free optimistic algorithm (quorum model)
  - 4 copies out of 6 needed for writes (4/6 write quorum - data considered durable when at least 4/6 copies acknowledge the write)
  - 3 copies out of 6 needed for reads (3/6 read quorum)
  - Self healing with peer-to-peer replication, Storage is striped across 100s of volumes
- One DocumentDB Instance takes writes (master)
- Compute nodes on replicas do not need to write/replicate (=improved read performance)
- Log-structured distributed storage layer – passes incremental log records from compute to storage layer (=faster)
- Master + up to 15 Read Replicas serve reads
- Data is continuously backed up to S3 in real time, using storage nodes (compute node performance is unaffected)

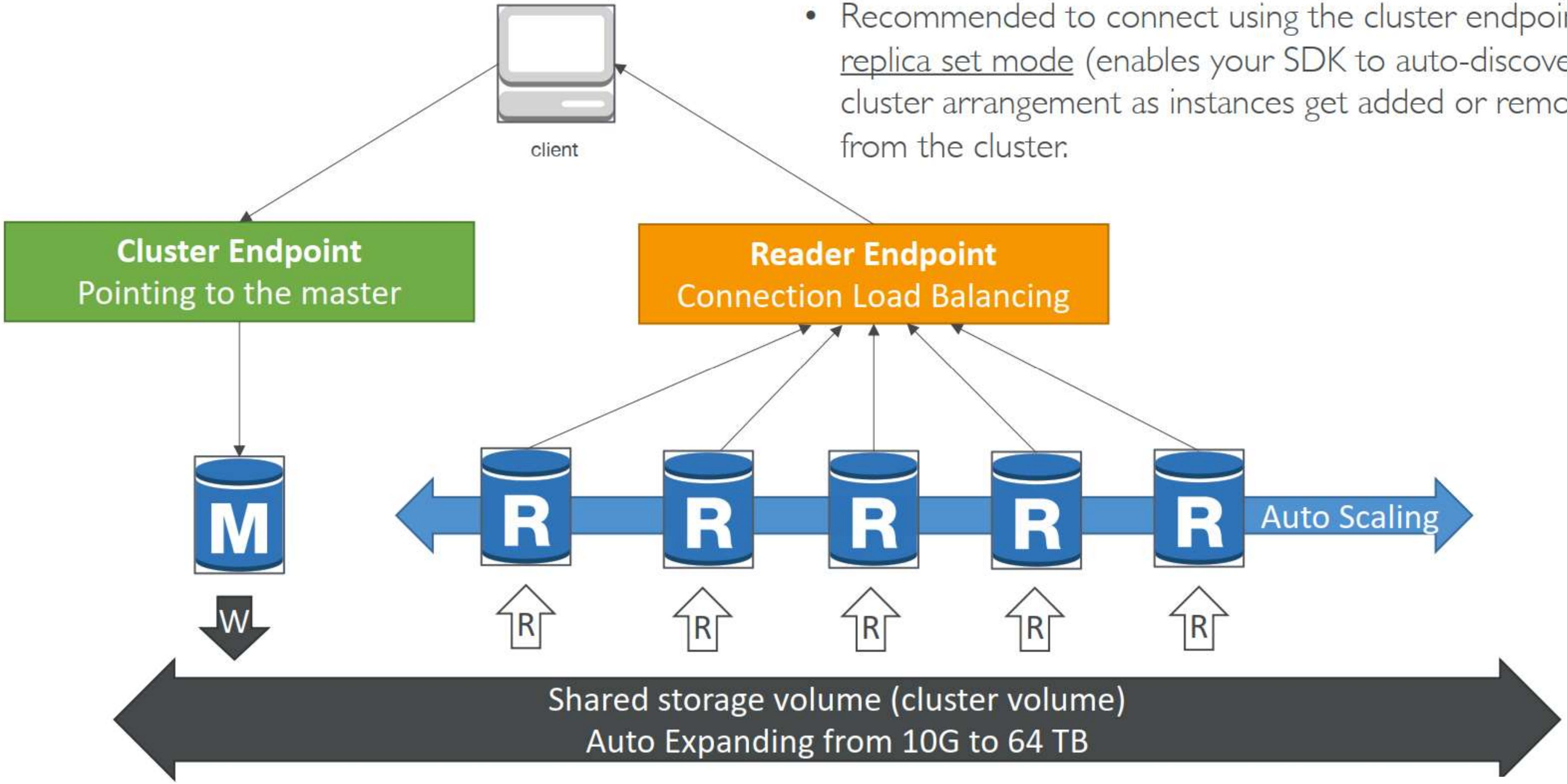
Same as Aurora architecture



Shared storage volume (cluster volume)
Replication + Self Healing + Auto Expanding

DocumentDB Cluster

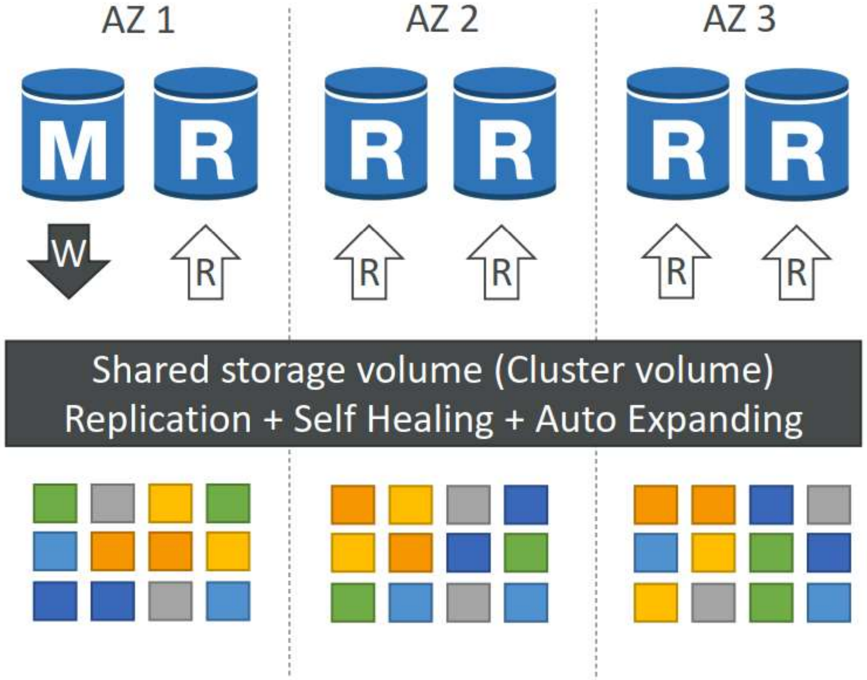
- Recommended to connect using the cluster endpoint in replica set mode (enables your SDK to auto-discover the cluster arrangement as instances get added or removed from the cluster.


client
Cluster Endpoint
Pointing to the master
Reader Endpoint
Connection Load Balancing
M
R
R
R
R
R
Auto Scaling
W
R
R
R
R
R
Shared storage volume (cluster volume)
Auto Expanding from 10G to 64 TB

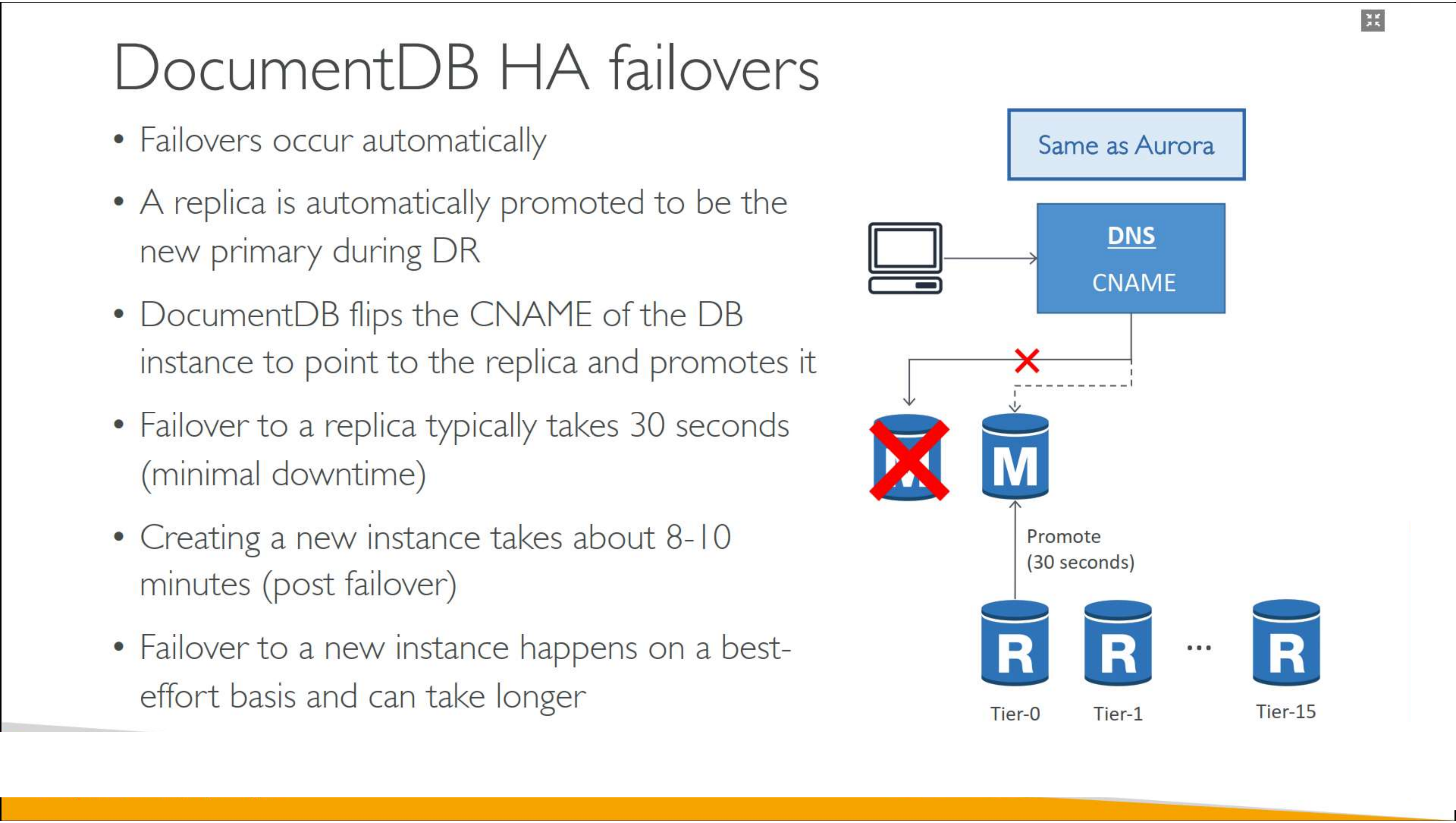
DocumentDB Replication

- Up to 15 read replicas
- ASYNC replication
- Replicas share the same underlying storage layer
- Typically take 10s of milliseconds (replication lag)
- Minimal performance impact on the primary due to replication process
- Replicas double up as failover targets (standby instance is not needed)

Same as Aurora



Shared storage volume (Cluster volume)
Replication + Self Healing + Auto Expanding



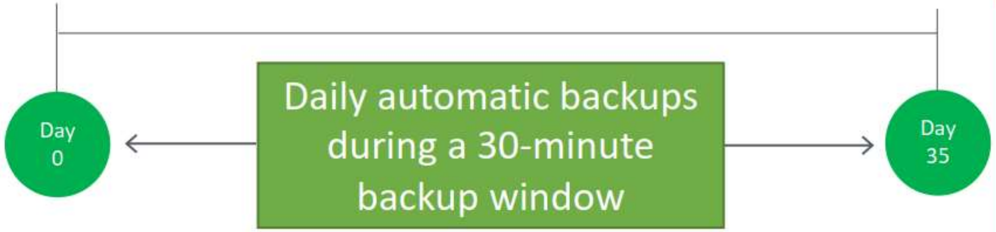
DocumentDB HA failovers

- Failovers occur automatically
- A replica is automatically promoted to be the new primary during DR
- DocumentDB flips the CNAME of the DB instance to point to the replica and promotes it
- Failover to a replica typically takes 30 seconds (minimal downtime)
- Creating a new instance takes about 8-10 minutes (post failover)
- Failover to a new instance happens on a best-effort basis and can take longer

DocumentDB Backup and Restore

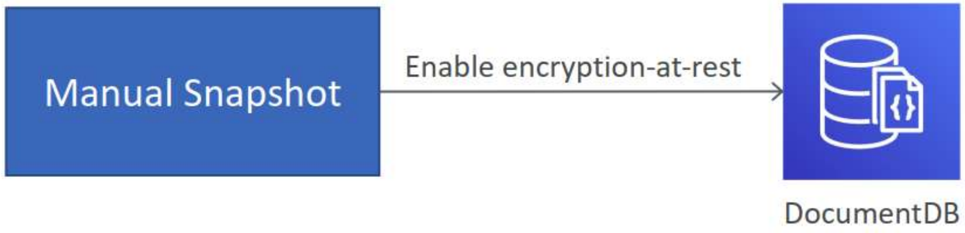
- Supports automatic backups
- Continuously backs up your data to S3 for PITR (max retention period of 35 days)
- latest restorable time for a PITR can be up to 5 mins in the past
- The first backup is a full backup. Subsequent backups are incremental
- Take manual snapshots to retain beyond 35 days
- Backup process does not impact cluster performance

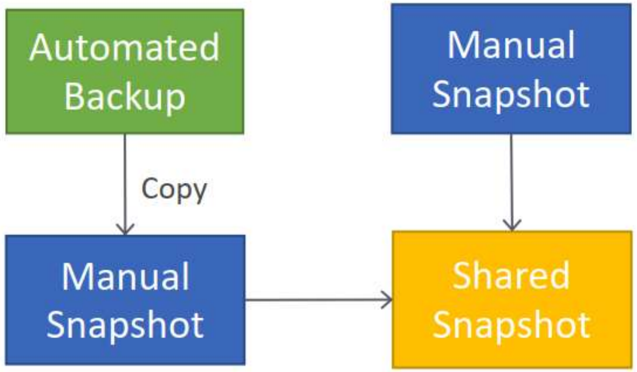
Same as RDS



DocumentDB Backup and Restore

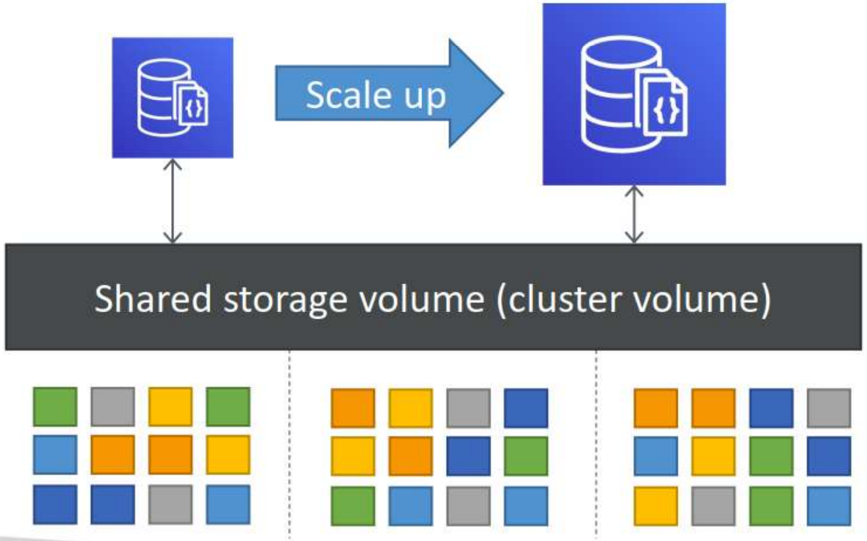
- Can only restore to a new cluster
- Can restore an unencrypted snapshot to an encrypted cluster (but not the other way round)
- To restore a cluster from an encrypted snapshot, you must have access to the KMS key
- Can only share manual snapshots (can copy and share automated ones)
- Can't share a snapshot encrypted using the default KMS key of the a/c
- Snapshots can be shared across accounts, but within the same region

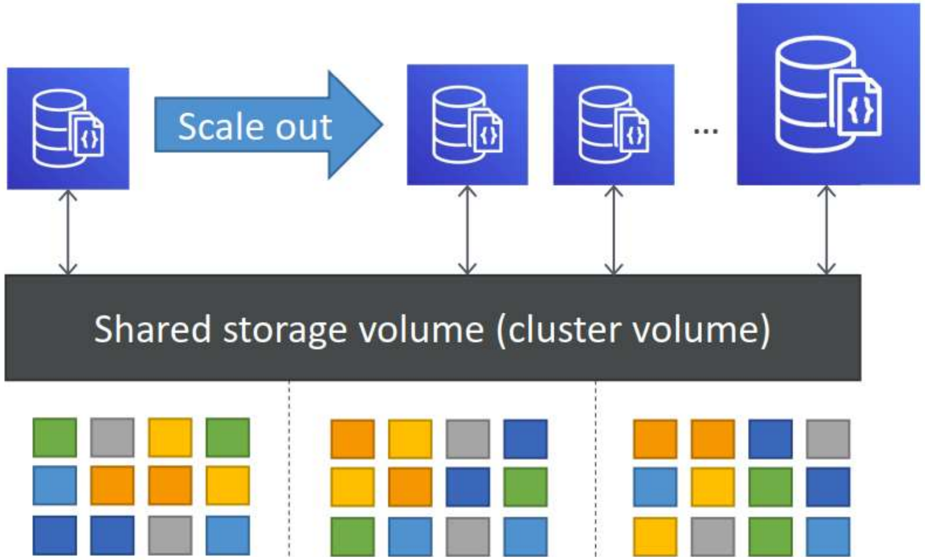




DocumentDB Scaling

- MongoDB sharding not supported (instead offers read replicas / vertical scaling / storage scaling)
- Vertical scaling (scale up / down) – by resizing instances
- Horizontal scaling (scale out / in) – by adding / removing up to 15 read replicas
- Can scale up a replica independently from other replicas (typically for analytical workloads)
- Automatic scaling storage – 10 GB to 64 TB (no manual intervention needed)

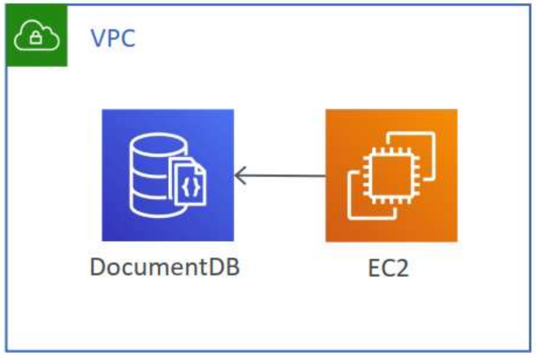




DocumentDB Security – IAM & Network

- You use IAM to manage DocumentDB resources
- Supports MongoDB default auth SCRAM (Salted Challenge Response Authentication Mechanism) for DB authentication
- Supports built-in roles for DB users with RBAC (role-based access control)
- DocumentDB clusters are VPC-only (use private subnets)
- Clients (MongoDB shell) can run on EC2 in public subnets within VPC
- Can connect to your on-premises IT infra via VPN

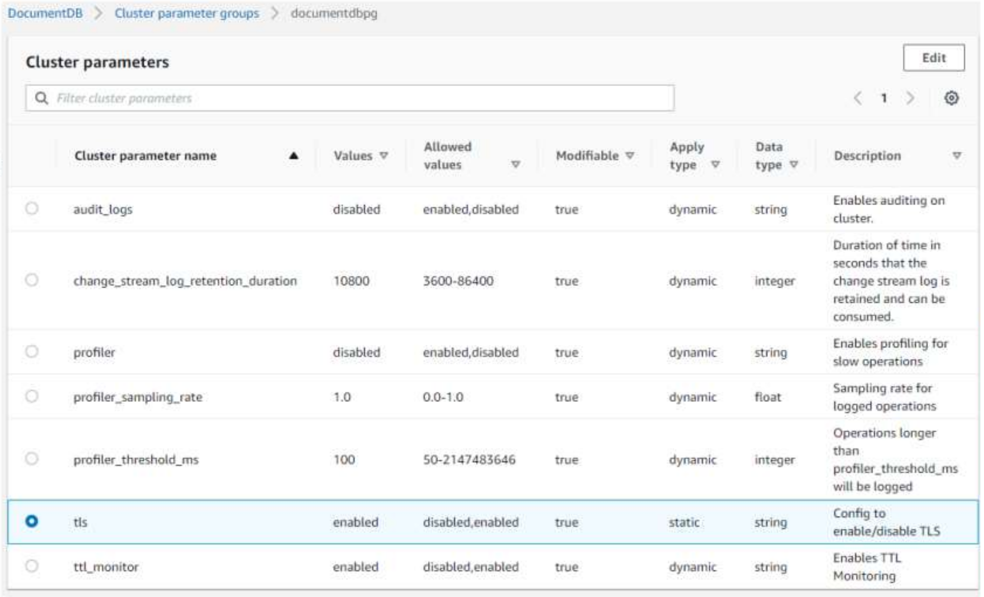
IAM



DocumentDB Security – Encryption

- Encryption at rest – with AES-256 using KMS
  - Applied to cluster data / replicas / indexes / logs / backups / snapshots
- Encryption in transit – using TLS
  - To enable TLS, set **tls** parameter in cluste parameter group
- To connect over TLS:
  - Download the certificate (public key) from AWS
  - Pass the certificate key while connecting to the cluster

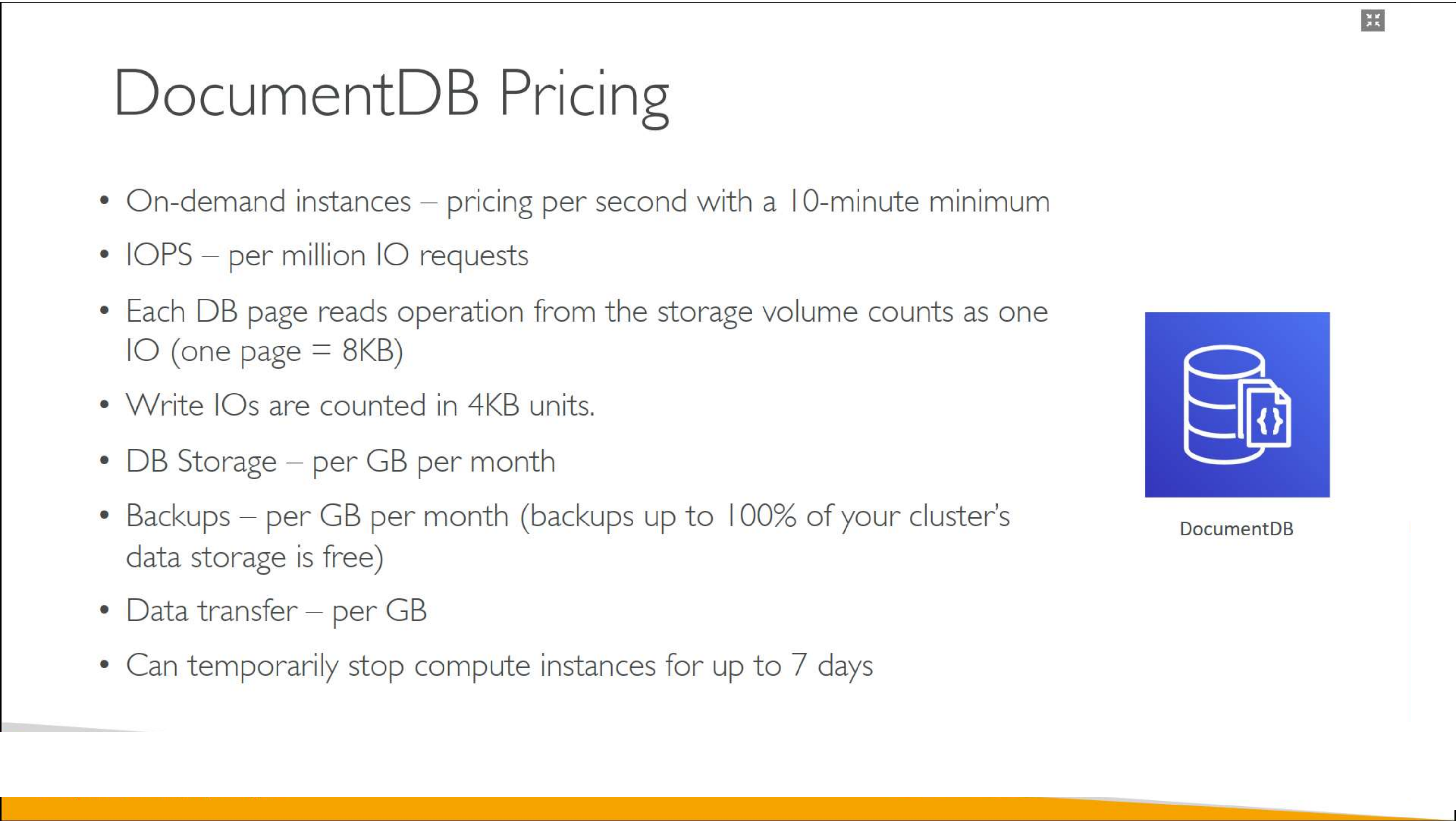
KMS



DocumentDB > Cluster parameter groups > documentdbpg

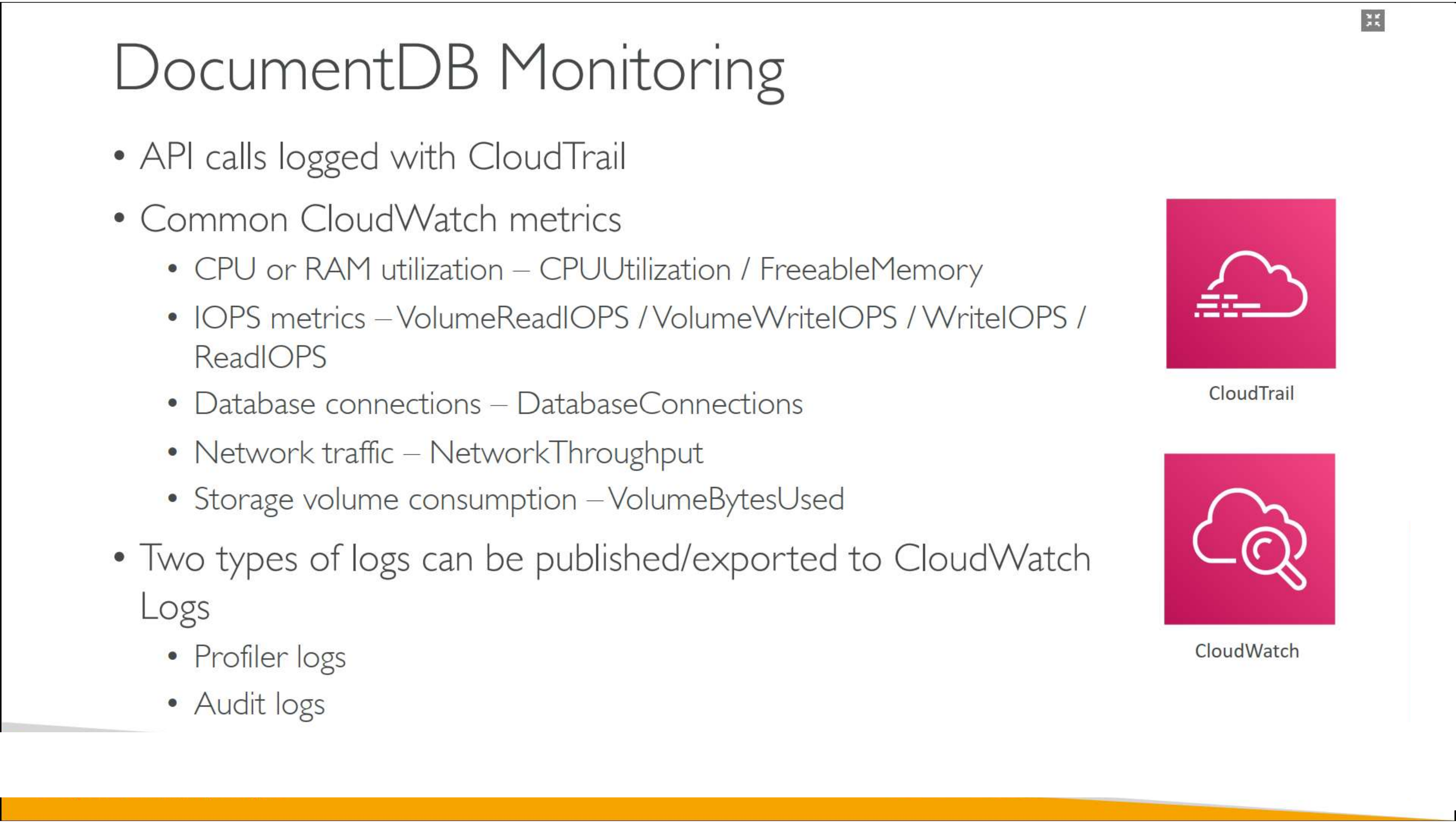
Cluster parameters

	Cluster parameter name	Values	Allowed values	Modifiable	Apply type	Data type	Description
○	audit_logs	disabled	enabled,disabled	true	dynamic	string	Enables auditing on cluster.
○	change_stream_log_retention_duration	10800	3600-86400	true	dynamic	integer	Duration of time in seconds that the change stream log is retained and can be consumed.
○	profiler	disabled	enabled,disabled	true	dynamic	string	Enables profiling for slow operations
○	profiler_sampling_rate	1.0	0.0-1.0	true	dynamic	float	Sampling rate for logged operations
○	profiler_threshold_ms	100	50-2147483646	true	dynamic	integer	Operations longer than profiler_threshold_ms will be logged
●	tls	enabled	disabled,enabled	true	static	string	Config to enable/disable TLS
○	ttl_monitor	enabled	disabled,enabled	true	dynamic	string	Enables TTL Monitoring



DocumentDB Pricing

- On-demand instances – pricing per second with a 10-minute minimum
- IOPS – per million IO requests
- Each DB page reads operation from the storage volume counts as one IO (one page = 8KB)
- Write IOs are counted in 4KB units.
- DB Storage – per GB per month
- Backups – per GB per month (backups up to 100% of your cluster's data storage is free)
- Data transfer – per GB
- Can temporarily stop compute instances for up to 7 days



DocumentDB Monitoring

- API calls logged with CloudTrail
- Common CloudWatch metrics
  - CPU or RAM utilization – CPUUtilization / FreeableMemory
  - IOPS metrics – VolumeReadIOPS / VolumeWriteIOPS / WriteIOPS / ReadIOPS
  - Database connections – DatabaseConnections
  - Network traffic – NetworkThroughput
  - Storage volume consumption – VolumeBytesUsed
- Two types of logs can be published/exported to CloudWatch Logs
  - Profiler logs
  - Audit logs

CloudTrail

CloudWatch

DocumentDB Profiler (profiler logs)

- Logs (into CloudWatch Logs) the details of ops performed on your cluster
- Helps identify slow operations and improve query performance
- Accessible from CloudWatch Logs
- To enable profiler:
  - Set the parameters – profiler, profiler_threshold_ms, and profiler_sampling_rate
  - Enable Logs Exports for Audit logs by modifying the instance
  - Both the steps above are mandatory

CloudWatch

DocumentDB audit logs

- Records DDL statements, authentication, authorization, and user management events to CloudWatch Logs
- Exports your cluster's auditing records (JSON documents) to CloudWatch Logs
- Accessible from CloudWatch Logs
- To enable auditing:
  - Set parameter audit_logs=enabled
  - Enable Logs Exports for Audit logs by modifying the instance
  - Both the steps above are mandatory

CloudWatch

DocumentDB Performance Management

- Use explain command to identify slow queries

```
db.runCommand({explain: {<query document>}})
```

- Can use **db.adminCommand** to find and terminate queries
- Example – to terminate long running / blocked queries

```
db.adminCommand({killOp: 1, op: <opid of the query>});
```

Disaster Recovery

Overview of Amazon DocumentDB Global Clusters

One primary region and up to five read-only secondary regions

Automatically replicates the data to the secondary regions

Benefits

- Recovery from region-wide outages
- Global reads with local latency
- Scalable secondary clusters
- Fast replication from primary to secondary clusters

Limitations

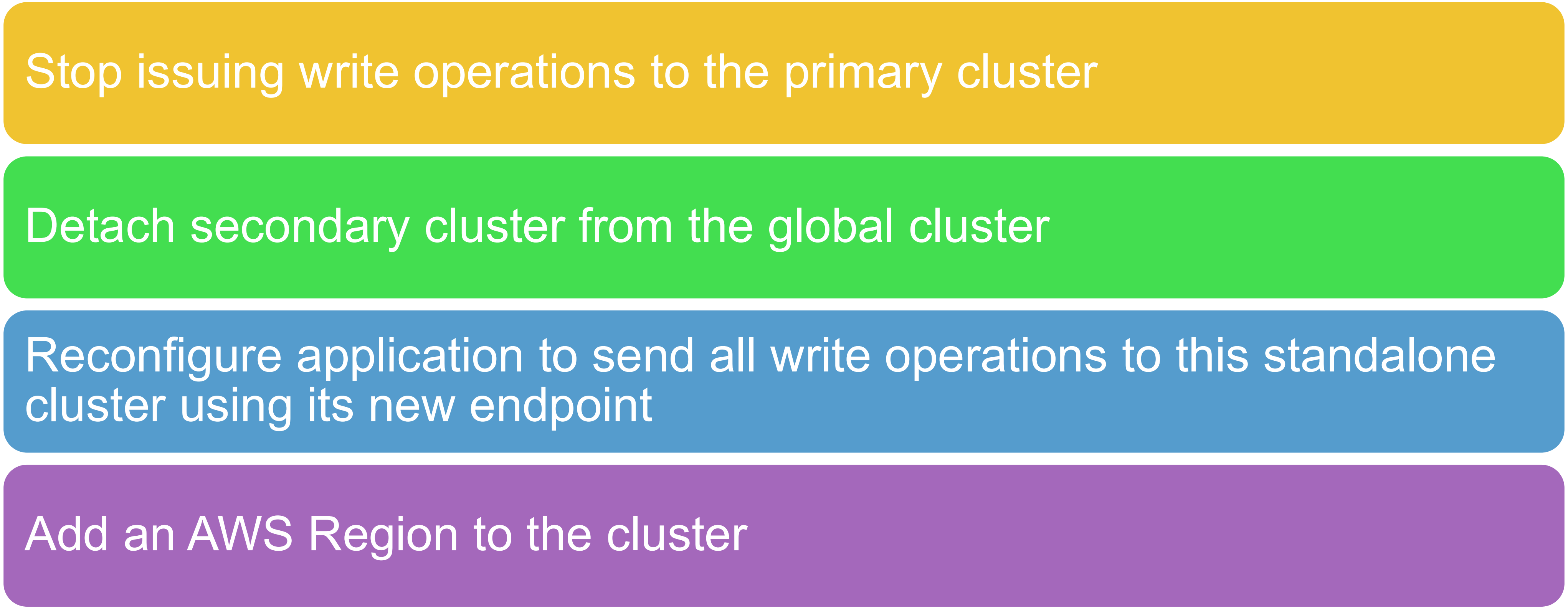
- Not supported on Amazon DocumentDB v3.6
- Not available in the following regions:
  - AWS GovCloud (US-West),
  - South America (São Paulo),
  - Europe (Milan),
  - China
  - China
- Must manually promote a secondary cluster
- Maximum of five secondary regions
- A primary cluster cannot be stopped if it has secondary clusters associated with it

Failover for Amazon DocumentDB Global Clusters

If an entire cluster in one AWS Region becomes unavailable, can promote another cluster to have read/write capability.

Can manually activate the failover mechanism if a cluster in a different AWS Region is a better choice to be the primary cluster.

Steps to promote secondary cluster



Stop issuing write operations to the primary cluster

Detach secondary cluster from the global cluster

Reconfigure application to send all write operations to this standalone cluster using its new endpoint

Add an AWS Region to the cluster

- Make sure that application writes are sent to the correct cluster before, during, and after making changes such as these, to avoid data inconsistencies among the clusters in the global cluster

Resilience in Amazon DocumentDB

Can design and operate applications and databases that automatically fail over between AZs without interruption

Cluster volume for DocumentDB cluster always spans 3 AZs to provide durable storage with less possibility of data loss.

Each 10 GB portion of your storage volume is replicated six ways, across three Availability Zones

Uses fault-tolerant storage that transparently handles

- The loss of up to two copies of data without affecting database write availability, and
- Up to three copies without affecting read availability

Self-healing;

- Data blocks and disks are continuously scanned for errors and replaced automatically

Manual backups and restore

Point-in-time recovery

User administration

Authenticating with identities

Authentication is how you sign in to AWS using your identity credentials

You must be authenticated (signed in to AWS)

Managing access using policies

Control access in AWS by creating policies and attaching them to AWS identities

A policy is an object in AWS that, when associated with an identity or resource, defines their permissions

Permissions in the policies determine whether the request is allowed or denied.

Policies are stored in AWS as JSON documents

To grant users permission, create IAM policies and add policies to users

Identity-based policy examples for Amazon DocumentDB
- Allow users to view their own permissions

```
{
  "Version": "2012-10-17",
  "Statement": [
      {
          "Sid": "ViewOwnUserInfo",
          "Effect": "Allow",
          "Action": [
              "iam:GetUserPolicy",
              "iam:ListGroupsForUser",
              "iam:ListAttachedUserPolicies",
              "iam:ListUserPolicies",
              "iam:GetUser"
          ],
          "Resource": ["arn:aws:iam::*:user/${aws:username}"]
      },
      {
          "Sid": "NavigateInConsole",
          "Effect": "Allow",
          "Action": [
              "iam:GetGroupPolicy",
              "iam:GetPolicyVersion",
              "iam:GetPolicy",
              "iam:ListAttachedGroupPolicies",
              "iam:ListGroupPolicies",
              "iam:ListPolicyVersions",
              "iam:ListPolicies",
              "iam:ListUsers"
          ],
          "Resource": "*"
      }
  ]
}
```

Managing Access Permissions to Your Amazon DocumentDB Resources

In Amazon DocumentDB, the primary resource is a cluster

Amazon DocumentDB supports other resources that can be used with the primary resource such as instances, parameter groups, and event subscriptions

These resources are referred to as subresources.

These resources and subresources have unique Amazon Resource Names (ARNs)

Managing Access Permissions to Your Amazon DocumentDB Resources

Resource Type	**ARN Format**
Cluster	arn:aws:rds:region:account-id:cluster:db-cluster-name
Cluster parameter group	arn:aws:rds:region:account-id:cluster-pg:cluster-parameter-group-name
Cluster snapshot	arn:aws:rds:region:account-id:cluster-snapshot:cluster-snapshot-name
Instance	arn:aws:rds:region:account-id:db:db-instance-name
Security group	arn:aws:rds:region:account-id:secgrp:security-group-name
Subnet group	arn:aws:rds:region:account-id:subgrp:subnet-group-name

Managing Access Permissions to Your Amazon DocumentDB Resources

- Example

```
• {
•     "Version": "2012-10-17",
•     "Statement": [
•         {
•             "Sid": "AllowCreateDBInstanceOnly",
•             "Effect": "Allow",
•             "Action": [
•                 "rds:CreateDBInstance"
•             ],
•             "Resource": [
•                 "arn:aws:rds:*:123456789012:db:test*",
•                 "arn:aws:rds:*:123456789012:pg:cluster-pg:default*",
•                 "arn:aws:rds:*:123456789012:subgrp:default"
•             ]
•         }
•     ]
• }
```

Using Identity-Based Policies (IAM Policies) for Amazon DocumentDB

- Allow a User to Perform Any Describe Action on Any Amazon DocumentDB Resource

```
• {
•    "Version":"2012-10-17",
•    "Statement":[
•       {
•          "Sid":"AllowRDSDescribe",
•          "Effect":"Allow",
•          "Action":"rds:Describe*",
•          "Resource":"*"
•       }
•    ]
• }
```

Using Identity-Based Policies (IAM Policies) for Amazon DocumentDB

- Prevent a User from Deleting an Instance

```
• {
•   "Version":"2012-10-17",
•   "Statement":[
•      {
•         "Sid":"DenyDelete1",
•         "Effect":"Deny",
•         "Action":"rds:DeleteDBInstance",
•         "Resource":"arn:aws:rds:us-east-1:123456789012:db:my-db-instance"
•      }
•   ]
• }
```

Using Identity-Based Policies (IAM Policies) for Amazon DocumentDB

•Prevent a User from Creating a Cluster unless Storage Encryption is Enabled

```
• {
•   "Version": "2012-10-17",
•   "Statement": [
•     {
•       "Sid": "PreventUnencryptedDocumentDB",
•       "Effect": "Deny",
•       "Action": "RDS:CreateDBCluster",
•       "Condition": {
•       "Bool": {
•       "rds:StorageEncrypted": "false"
•     },
•       "StringEquals": {
•       "rds:DatabaseEngine": "docdb"
•       }
•     },
•     "Resource": "*"
•     }
•   ]
• }
```

AWS managed policies for Amazon DocumentDB

AmazonDocDBFullAccess

- Grants administrative permissions

AmazonDocDBReadOnlyAccess

- Grants read-only permissions that allow users to view information in Amazon DocumentDB

AmazonDocDBConsoleFullAccess

- Grants full access to manage Amazon DocumentDB resources using the AWS Management Console

AmazonDocDBElasticReadOnlyAccess

- Grants read-only permissions that allow users to view elastic cluster information

AmazonDocDBElasticFullAccess

- Grants administrative permissions that allow a principal full access to all Amazon DocumentDB actions for Amazon DocumentDB elastic cluster.

Backup and Restore

Backing Up and Restoring in Amazon DocumentDB

- Continuously backs up your data to Amazon Simple Storage Service (Amazon S3) for 1–35 days

Prerequisites

- wget https://fastdl.mongodb.org/tools/db/mongodb-database-tools-ubuntu2204-x86_64-100.9.0.deb
- sudo apt install ./mongodb-database-tools-*-100.9.0.deb
- wget https://truststore.pki.rds.amazonaws.com/global/global-bundle.pem
- wget https://raw.githubusercontent.com/ozlerhakan/mongodb-json-files/master/datasets/restaurant.json
- mongoimport --ssl --host="ag-docdb-2023-10-31-06-30-19.cluster-cmihymqkhemz.us-east-1.docdb.amazonaws.com:27017" --collection=restaurants --db=business --file=restaurant.json --numInsertionWorkers 4 --username=atingupta2005 --password=Aws123456 --sslCAFile global-bundle.pem

Dumping

-
```
mongodump --ssl --host="ag-docdb-2023-10-31-06-30-19.cluster-cmihymqkhemz.us-east-1.docdb.amazonaws.com:27017" --collection=restaurants --db=business --out=restaurantDump.bson --numParallelCollections 4 --username=atingupta2005 --password=Aws123456 --sslCAFile global-bundle.pem
```

Drop

- mongosh --ssl --host ag-docdb-2023-10-31-06-30-19.cluster-cmihymqkhemz.us-east-1.docdb.amazonaws.com:27017 --sslCAFile global-bundle.pem --username atingupta2005 --password Aws123456

- use business

- db.restaurants.drop()

Restore

```
•mongorestore --ssl --host="ag-docdb-2023-10-31-06-30-19.cluster-cmihymqkhemz.us-east-1.docdb.amazonaws.com:27017" --numParallelCollections 4 --username=atingupta2005 --password=Aws123456 --sslCAFile global-bundle.pem restaurantDump.bson
```

Export

```
mongoexport --ssl --host="ag-docdb-2023-10-31-06-30-19.cluster-cmihymqkhemz.us-east-1.docdb.amazonaws.com:27017" --collection=restaurants --db=business --out=restaurant2.json --username=atingupta2005 --password=Aws123456 --sslCAFile global-bundle.pem
```

Validation

- wc -l restaurant.json
- tail restaurant.json
- head restaurant.json

Cluster Snapshot

- Amazon DocumentDB creates daily automatic snapshots
- Can also manually create a cluster snapshot
- Backup retention period of 1–35 days

Creating a Cluster Snapshot

- Hands-on

Restoring from a Cluster Snapshot

- Hands-on

Restoring to a Point in Time

- Hands-on

Deleting a Cluster Snapshot

- Hands-on

Amazon DocumentDB High Availability and Replication

Read Scaling

Amazon DocumentDB replicas work well for read scaling

Write operations are managed by the primary instance

The cluster volume is shared among all instances in your cluster

- Therefore, you don't have to replicate and maintain a copy of the data for each Amazon DocumentDB replica.

High Availability

When you create an Amazon DocumentDB cluster, it provisions instances across the Availability Zones.

When you create instances in the cluster, Amazon DocumentDB automatically distributes the instances across the Availability Zones in a subnet group to balance the cluster

This action also prevents all instances from being located in the same Availability Zone.

Adding Replicas

- The first instance added to the cluster is the primary instance
- Every instance that is added after the first instance is a replica instance
- A cluster can have up to 15 replica instances in addition to the primary.
- When you create a cluster using the AWS Management Console, a primary instance is automatically created at the same time
- To create a replica at the same time as you create the cluster and the primary instance, choose Create replica in different zone

Replication Lag

Is typically 50ms or less

The most common reasons for increased replica lag are:

- A high write rate on the primary that causes the read replicas to fall behind the primary.
- Contention on the read replicas between long running queries (e.g., large sequential scans, aggregation queries) and incoming write replication.
- Very large number of concurrent queries on the read replicas.

To minimize replication lag

- If you have a high write rate or high CPU utilization, we recommend that you scale up the instances in your cluster.
- If there are long running queries on your read replicas, and very frequent updates to the documents being queried, consider altering your long running queries, or running them against the primary/write replica to avoid contention on the read replicas.
- If there is a very large number of concurrent queries or high CPU utilization only on the read replicas, scale out the number of read replicas to spread out the workload.

DocumentDB Elastic Clusters

What is Amazon DocumentDB Elastic Clusters?

Amazon DocumentDB Elastic Clusters enables to elastically scale your document database to handle millions of writes and reads, with petabytes of storage capacity

Automatically manage the underlying infrastructure

No need to create, remove, upgrade, or scale instances

How do I get started with Elastic Clusters?

You can create an Elastic Clusters cluster

- Using DocumentDB API, SDK, CLI, CloudFormation (CFN), or the AWS console

When provisioning your cluster, specify how many shards and the compute per shard that the workload needs.

Once created cluster, start leveraging Elastic Clusters' elastic scalability.

Depending on the workload's needs, can add or remove compute by modifying shard count and/or compute per shard

Elastic Clusters will automatically provision/de-provision the underlying infrastructure and rebalance data

How does Elastic Clusters work?

Elastic Clusters uses sharding to partition data across Amazon DocumentDB’s distributed storage system

Sharding, also known as partitioning, splits large data sets into small data sets across multiple nodes enabling customers to scale out their database beyond vertical scaling limits of a single database

Elastic Clusters utilizes the separation of compute and storage in Amazon DocumentDB

Rather than re-partitioning collections by moving small chunks of data between compute nodes, Elastic Clusters can copy data efficiently within the distributed storage system.

What types of sharding does Elastic Clusters support?

- Hash-based partitioning.

How is Elastic Clusters different from MongoDB sharding?

With Elastic Clusters, you can easily scale out or scale in your workload on Amazon DocumentDB typically with little to no application downtime or impact to performance regardless of data size

A similar operation on MongoDB would impact application performance and take hours, and in some cases days.

Elastic Clusters also offers differentiated management capabilities such as no impact backups and rapid point in time restore

How do I define a shard key?

- The ideal shard key distributes data evenly across the sharded cluster while also facilitating common query patterns.

Concepts associated with Elastic Clusters

Elastic Clusters

- An Amazon DocumentDB cluster that allows you to scale your workload's throughput to millions of reads/writes per second and storage to petabytes
- An Elastic Cluster cluster comprises of one or more shards for compute and a storage volume, and is highly available across multiple Availability Zones by default.

Shard

- A shard provides compute for Elastic Clusters cluster
- A shard by default will have three nodes, one writer node and two reader nodes
- You can have a maximum of 32 shards and each shard can have a maximum of 64 vCPUs.

Shard key

- Shard key is an optional field in your JSON documents that Elastic Clusters uses to distribute read and write traffic to the matching shard.
- You are advised to pick a key that has lots of unique values. A good shard key will evenly partition your data across the underlying shards, giving your workload the best throughput and performance.

Sharded collection

- A collection whose data is distributed across an Elastic Clusters cluster.

Best practices

Basic Operational Guidelines

Deploy a cluster consisting of two or more Amazon DocumentDB instances in two AWS Availability Zones

Use the service within the stated service limits

Monitor your memory, CPU, connections, and storage usageS

- Set up Amazon CloudWatch to notify

Scale up instances when you are approaching capacity limits

Set your backup retention period to align with your recovery point objective.

Test failover for your cluster to understand how long the process takes for your use case

Connect to your Amazon DocumentDB cluster with the cluster endpoint

Choose a driver read preference setting that maximizes read scaling while meeting your application's read consistency requirements.

- primary
- primaryPreferred
- secondary
- secondaryPreferred
- nearest

Design application to be resilient in the event of network and database errors

Enable cluster deletion protection for all production clusters

Instance Sizing

One of the most critical aspects of choosing an instance size in DocumentDB is the amount of RAM for cache

DocumentDB reserves one-third of the RAM for its own services

Only two-thirds of the instance RAM is available for the cache

DocumentDB best practice to choose an instance type with enough RAM to fit your working set (i.e., data and indexes) in memory

Having properly sized instances will help optimize for overall performance and potentially minimize I/O cost

To determine whether application's working set fits in memory, monitor the BufferCacheHitRatio using CloudWatch for each instance

The BufferCacheHitRatio CloudWatch metric measures the percentage of data and indexes served from an instance’s memory cache

Working with Indexes

When importing data into Amazon DocumentDB, should create indexes before importing large datasets

Limit the creation of indexes to fields where the number of duplicate values is less than 1%

- As an example, if your collection contains 100,000 documents, only create indexes on fields where the same value occurs 1000 times or fewer.

Only create indexes on fields that are commonly utilized as a filter

Regularly look for unused indexes

Cost Optimization

- Create billing alerts at thresholds
- Can optimize costs by using a single instance development cluster when high availability is not required.
- For development and test scenarios, stop a cluster when it is no longer needed

Using Metrics to Identify Performance Issues

Viewing Performance Metrics

Setting a CloudWatch Alarm

Evaluating Performance Metrics

- CPU Utilization
- Freeable Memory
- Swap Usage
- Read IOPS, Write IOPS
- Read Latency, Write Latency
- Read Throughput, Write Throughput
- Disk Queue Depth
- Network Receive Throughput, Network Transmit Throughput
- DB Connections

Recommendations and advice about specific types of metrics

High CPU consumption

- If your CPU consumption is consistently over 80 percent, consider scaling up your instances.

High RAM consumption

- If your FreeableMemory metric frequently dips below 10% of the total instance memory, consider scaling up your instances.

Swap usage

- Should remain at or near zero. If your swap usage is significant, consider scaling up your instances.

Network traffic

- Investigate network traffic if throughput is consistently lower than expected.

batchInsert and batchUpdate

If FreeableMemory goes to zero on primary instance

- Either reduce the concurrency of the batch insert or
- Update workload or
- Increase the instance size to increase the amount of FreeableMemory.

Security aspects

Data Protection

Use multi-factor authentication (MFA) with each account

Use SSL/TLS to communicate with AWS resources

Set up API and user activity logging with AWS CloudTrail

Use AWS encryption solutions, along with all default security controls within AWS services

Use advanced managed security services such as Amazon Macie, which assists in discovering and securing sensitive data that is stored in Amazon S3

Use DocumentDB client-side field level encryption (FLE)

- Encrypt sensitive data in your client applications before it is transferred to a Amazon DocumentDB cluster.

Encrypting Amazon DocumentDB Data at Rest

Encrypting Data in Transit

Identity and Access Management

- Control who can be authenticated (signed in) and authorized (have permissions) to use Amazon DocumentDB resources.

Managing Users

- Hands-on

Database Access Using Role-Based Access Control

```
db.createUser({user: "sample-user", pwd: "abc123", roles: [{role: "read", db: "sample-database"}]})
show users
db.createUser({user: "user1", pwd: "abc123", roles: [{role: "readWriteAnyDatabase", db: "admin"}]})
db.createUser({user: "user2", pwd: "abc123", roles: [{role: "read", db: "sample-database-1"}]})
db.createUser({user: "user3", pwd: "abc123", roles: [{role: "readWrite", db: "sample-database-1"}]})
db.grantRolesToUser("user3", [{role: "read", db: "sample-database-2"}])
db.dropUser("user1")
db.dropUser("user2")
show users
db.updateUser("readInFooBar", {roles: [{role: "read", db: "foo"}, {role: "read", db: "baz"}]})
db.getRole("read", {showPrivileges:true})
```

Logging and Monitoring in Amazon DocumentDB

Can use the profiler to log the execution time and details of operations that were performed on your cluster

Profiler is useful for monitoring the slowest operations on your cluster to help you improve individual query performance and overall cluster performance

When enabled, operations are logged to Amazon CloudWatch Logs and you can use CloudWatch Insight to analyze, monitor, and archive your Amazon DocumentDB profiling data

Amazon DocumentDB also integrates with AWS CloudTrail, a a service that provides a record of actions taken by users, roles, or an AWS service in Amazon DocumentDB (with MongoDB compatibility).

With Amazon DocumentDB, you can audit events that were performed in your cluster

Examples of logged events include

- Successful and failed authentication attempts,
- Dropping a collection in a database, or
- Creating an index.

By default, auditing is disabled on Amazon DocumentDB and requires that you opt in to this feature.

Upgrading DocumentDB using DMS

- Hands-on

Thanks