

**Mc donalds** - Cloud migration is chosen as an cloud migration industry example.

**1) Mc Donalds** : It is one of leading fast food industry, renowned for its burgers. It has a huge network of around 37k restaurants in 120 countries, serves almost 69 million people per day. The main reason behind migration is the company can rent services for everything from apps to storage from AWS rather than having their own computing equipment or data centers.

One advantage of utilizing these cloud computing services is that businesses may avoid the upfront cost and complexity of building and maintaining their own IT infrastructure by just paying for what they need, when they use it.

## **2) Business drivers/ Goals behind the cloud migration:**

- **Digital transformation** - To make it more convenient as well as personalized to the customers by having a seamless experience.
- **Speed to market** - To Target of 250 - 500k orders per hour i.e 20,000 transactions per second.
- **Multi country support** - To integrate with multiple third party food delivery partners in different countries around the globe to incorporate the convenience theme

## **3) Technology Goals -**

Mc donalds chose Amazon Web Services native services for flexibility, self-service, and medium cost.

- To work on the limited scale, quick data demands and high fixed cost.
- Cloud model goal is to remove **Siloed data** with limited data availability and also helps to eliminate data movement.
- To use a variety of **analytical engines** to gain insight instead of having limited analysis focused.
- To have **Unified access** and governance.

#### 4) Migration strategy

To use micro service architecture for services like home delivery:

- All apis are hosted thru **API middleware** which is then wired through Application load balancer to ECS.
- Within ECS we have **multiple microservices** for illustration purposes.
- **Auto scale policies** will help to scale the microservices as per the requirement.
- For eventing or to communicate between services we use **SQS**,
- **Redis** is used as distributed caching platform to seamlessly hit the transaction volumes at about 100 ms or lower for each call.
- **RDS** is used for backing redis and **S3** for the organising the unstructured data.

#### 5) Organizational Change

- **Microservices** with cleaned API's, service models, isolation, independent data models and deployability.
- **Containers and orchestration**, for handling massive scale, reliability and speed to market requirements.
- **PaaS**, based architectural model by leveraging AWS platform components such as ECS, SQS, RDS and Elasticache.
- **Synchronous and event based**, programming models based on requirements.

#### 6) Cultural Change

- It started with individuals collecting orders at the counter and has evolved into a more tech-driven procedure that promises a quick and excellent dinner.
- McDonald's has only benefited from digital transformation in terms of speeding up service while maintaining the quality that their consumers expect.
- They've accomplished this by incorporating digital ordering kiosks, smartphone applications, and table service into some of their locations.

## 7) Migration Journey

- **Scalability and reliability**  
Achieved high scale targets using autoscaling , task placement strategies and perform more batch mode processing along with work load optimization.
- **Security** - Container isolation through IAM policies and security groups and reduced container-to container communication.
- **Monitoring** - Newrelic agents were configured to monitor ECS instances, containers and AWS Paas components.

## 8) Its a ongoing cloud journey and they achieved the following in the process:

- Integrated and Trusted data platform
- Descriptive, predictive and prescriptive analysis
- Self-service delivery model
- Data enables faster business insights and growth
- On demand scale with cloud, usage-based cost.

## 9) Lessons learnt

- According to Schneider, the company employed a "lift and shift" strategy to cloud migration, preserving identical code structure and constructing a new database avoiding duplicate functionality while making the necessary adjustments to business platforms.
- To complete the project, the organization collaborated with 70+ developers and 3 different consulting firms, learning several non-technical lessons along the way. However, bringing in outside expertise is always a benefit because consultants might experience a steep learning curve as well.

## 10) Outcomes achieved-

- A thought out microservice architecture is key for **scalability, reliability, and containerization**.
- **Massive scale achievable** (north of 20k TPS under 100ms) in a controlled manner using auto-scale policies and task placement strategies.

- **Moving to containers** simplified our development and deployment models and in turn provided quicker dev/test iterations.
- **ECS** out of box integration and deployment models further simplified the devops pipeline.

## **11) Cloud Migration Strategy:**

Things to consider in general for cloud migration:

1. Need to go through the business requirement and be sure on the problem we are targeting to solve by moving to cloud.
2. Need to assess the existing infrastructure and decide on the migration strategy. I would prefer to follow a mix and match of Rehost (Lift and Shift) and Refactor for the existing infrastructure.

Note: Other migration strategies include Revise, Rebuild and Replace.

3. Need to consider the security related aspects at application level.
4. Need to evaluate the cost included for the migration and then decide on the best provider for our use case.

Now, coming back to the MCD cloud migration strategy:

1. Creating infrastructure for PROD and COB in VPC's in two different regions and distributing the infrastructure into multiple AZ's, this enables the application to have high availability.
2. To handle the traffic coming from external applications we can deploy load balancers or reverse proxy.
3. For high available systems we should deploy application load balancers between PROD and COB, enabling Active-Active architecture.
4. As microservice are launched on to ECS, we should enable autoscaling. This can be done in two ways one is instance based autoscaling (which dynamically increase/decreases EC2 instances) or service autoscaling (which scales docker containers).
5. For AWS RDS, enabling multi-AZ will help switching to stand by instance in case of failure to the primary instance.
6. Need to implement IAM rules to secure and have control over all the resources.

## **Q2) Implemented below web scale cloud content storage and delivery infrastructure on AWS**

- a) To maximize the system's availability Amazon S3 buckets are been created.  
In my case, I have created four buckets in different regions respectively.**

The screenshot shows the 'Create bucket' wizard. In the 'General configuration' step, a bucket name 's3ptbucket1' is entered, and the AWS Region is set to 'US East (N. Virginia) us-east-1'. Under 'Object Ownership', the 'ACLs disabled (recommended)' option is selected, indicating that all objects in the bucket are owned by the account owner.

The screenshot shows the 'Buckets' page. It displays an 'Account snapshot' and a table of four buckets. The buckets are:

Name	AWS Region	Access	Creation date
s3ptbucket4	US West (N. California) us-west-1	Bucket and objects not public	September 21, 2022, 15:13:06 (UTC-07:00)
s3ptbucket3	Asia Pacific (Sydney) ap-southeast-2	Bucket and objects not public	September 21, 2022, 15:11:54 (UTC-07:00)
s3ptbucket2	EU (Paris) eu-west-3	Bucket and objects not public	September 21, 2022, 15:10:14 (UTC-07:00)
s3ptbucket1	US East (N. Virginia) us-east-1	Bucket and objects not public	September 21, 2022, 15:09:24 (UTC-07:00)

**b) Create CDN - It pulls data from an Amazon S3 bucket and distributes it across several datacenters. It distributes data via a network of data centers known as edge locations and have the below settings enabled while creating it.**

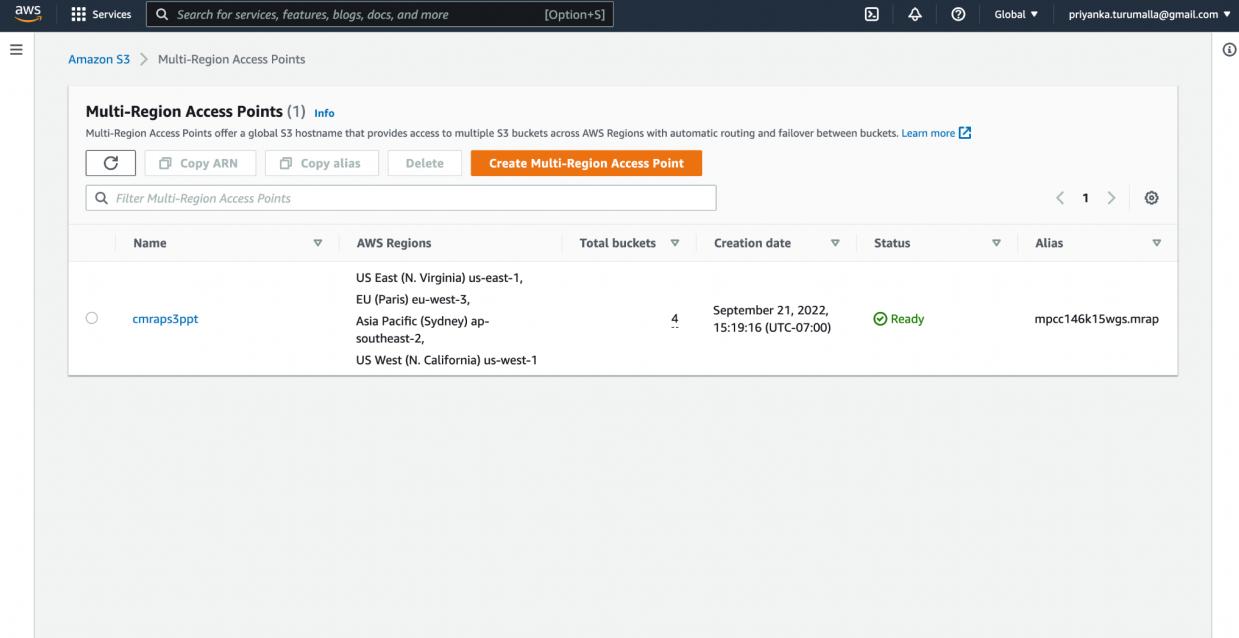
The screenshot shows the 'Edit Settings' page for a new CloudFront distribution. The 'Settings' tab is selected. Key configuration options visible include:

- Price class**: Set to "Use all edge locations (best performance)". Other options are "Use only North America and Europe" and "Use North America, Europe, Asia, Middle East, and Africa".
- AWS WAF web ACL**: A dropdown menu labeled "Choose web ACL".
- Alternate domain name (CNAME)**: A section for adding custom domain names, with a "Add item" button.
- Custom SSL certificate**: A dropdown menu labeled "Choose certificate" and a "Request certificate" button.
- Supported HTTP versions**: Options for enabling HTTP/2 and HTTP/3, with HTTP/2 checked by default.

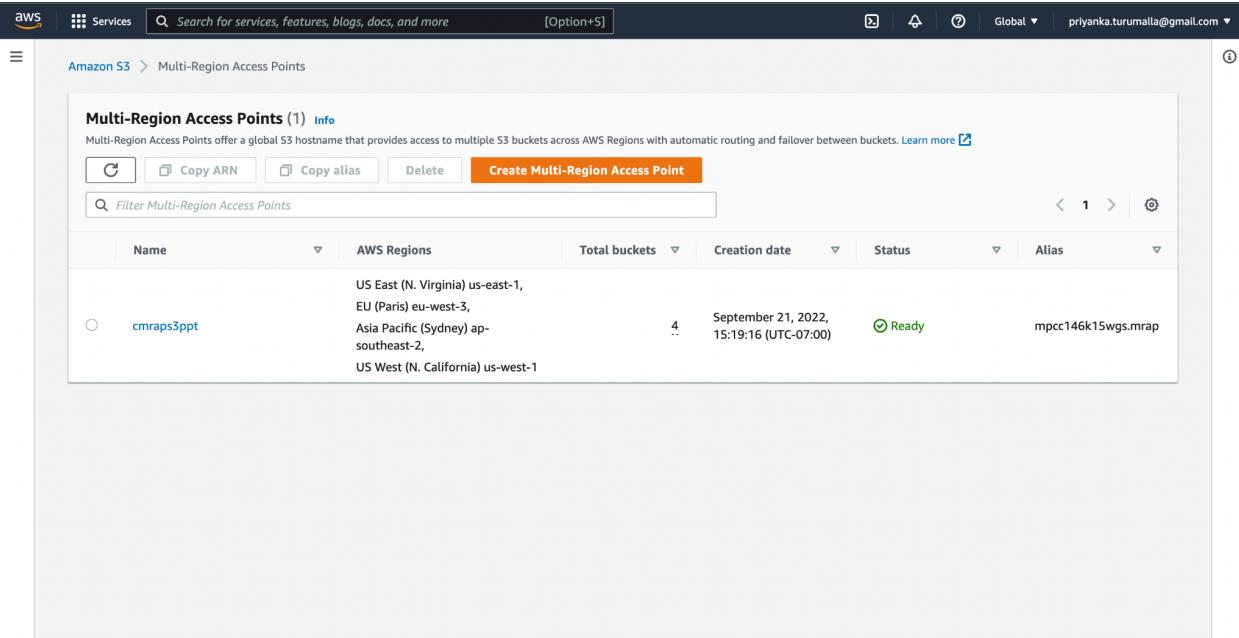
The screenshot shows the 'Distributions' page in the CloudFront console. It displays a single distribution entry:

ID	Description	Domain name	Alternate domai...	Origins	Status	Last modified
EUQIGJJ4BY745	-	d31u58h6jsv417.cl...	-	s3ptbucket1.s3.us-east-1.	Enabled	September 21, 202...

**c) Create multi- region access points to give access across the regions to multiple s3 buckets with automatic routing.**



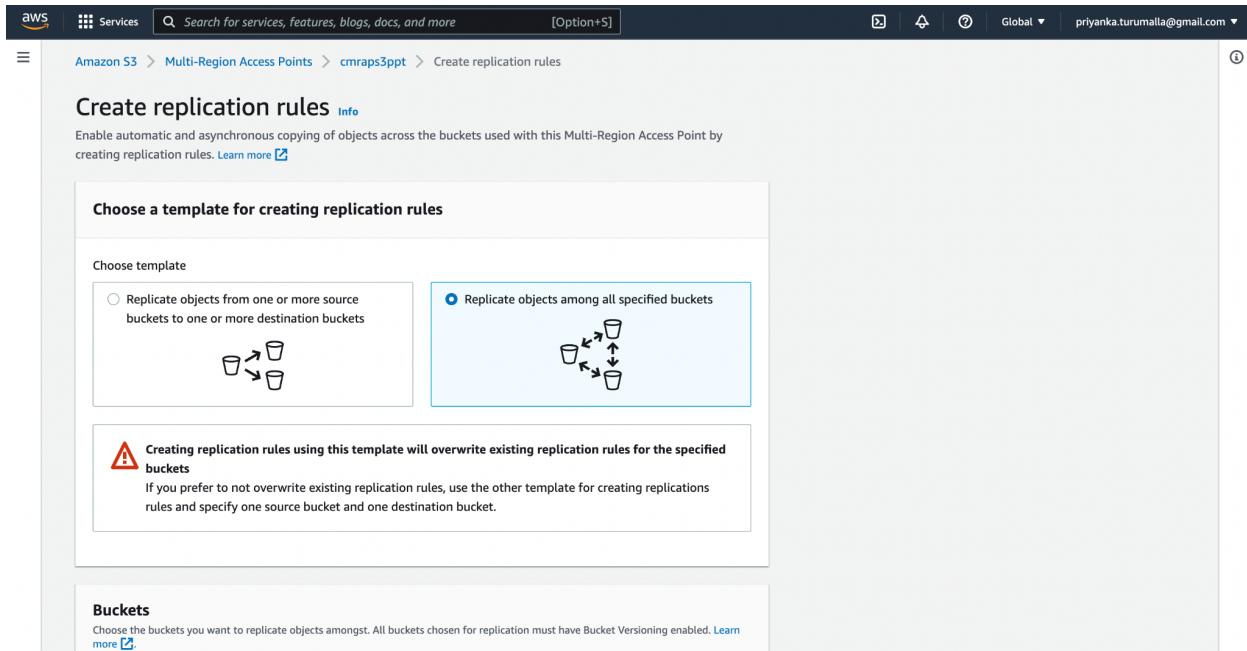
The screenshot shows the AWS S3 Multi-Region Access Points console. At the top, there is a search bar and a global navigation bar with the email 'priyanka.turumalla@gmail.com'. Below the header, the breadcrumb navigation shows 'Amazon S3 > Multi-Region Access Points'. The main area displays a table titled 'Multi-Region Access Points (1)'. The table has columns: Name, AWS Regions, Total buckets, Creation date, Status, and Alias. One entry is listed: 'cmraps3ppt' with 'US East (N. Virginia) us-east-1, EU (Paris) eu-west-3, Asia Pacific (Sydney) ap-southeast-2, US West (N. California) us-west-1' under 'AWS Regions', '4' under 'Total buckets', 'September 21, 2022, 15:19:16 (UTC-07:00)' under 'Creation date', 'Ready' under 'Status', and 'mpcc146k15wgs.mrap' under 'Alias'. There are buttons for 'Create Multi-Region Access Point', 'Copy ARN', 'Copy alias', and 'Delete'.



This screenshot is identical to the one above, showing the AWS S3 Multi-Region Access Points console with a single entry for 'cmraps3ppt'. The details are the same: 'cmraps3ppt' with 'US East (N. Virginia) us-east-1, EU (Paris) eu-west-3, Asia Pacific (Sydney) ap-southeast-2, US West (N. California) us-west-1' under 'AWS Regions', '4' under 'Total buckets', 'September 21, 2022, 15:19:16 (UTC-07:00)' under 'Creation date', 'Ready' under 'Status', and 'mpcc146k15wgs.mrap' under 'Alias'. The interface includes a search bar, a global navigation bar with the email 'priyanka.turumalla@gmail.com', and buttons for 'Create Multi-Region Access Point', 'Copy ARN', 'Copy alias', and 'Delete'.

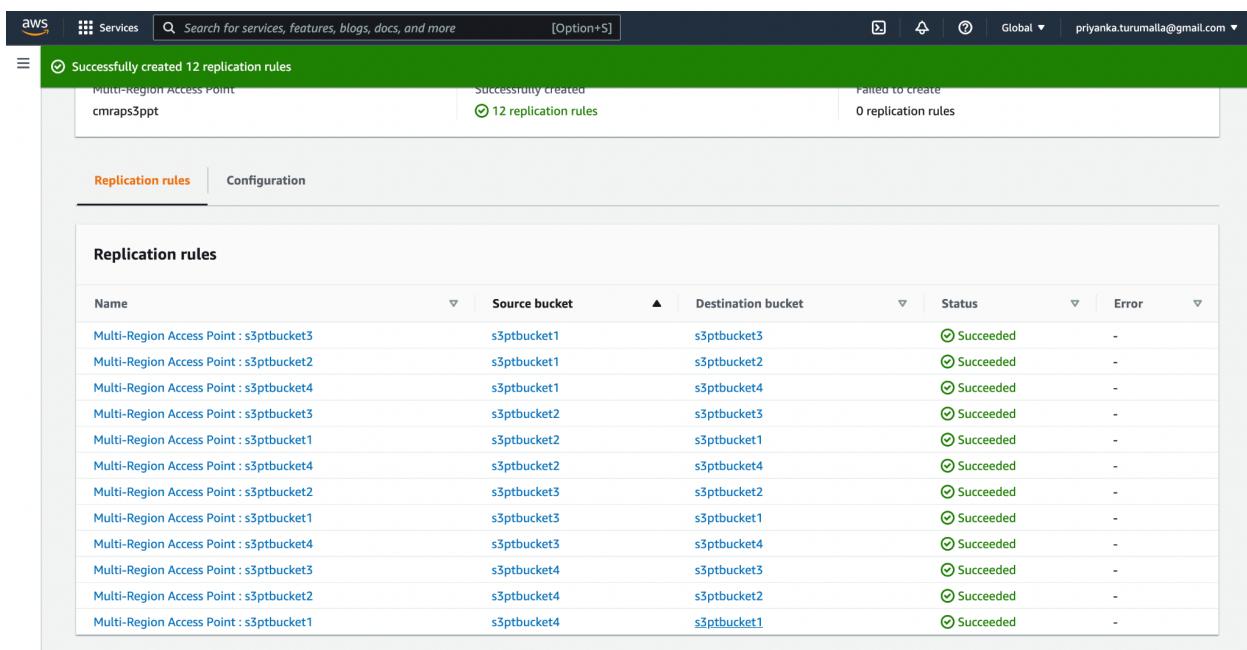
**d) To replicate the objects create the replication rules as a next step. This also helps as an ‘Disaster Recovery’.**

### Replication rules:



The screenshot shows the 'Create replication rules' page in the AWS S3 console. The top navigation bar includes the AWS logo, services menu, search bar, and user information. The main content area has a title 'Create replication rules' with an 'Info' link. A sub-instruction says 'Enable automatic and asynchronous copying of objects across the buckets used with this Multi-Region Access Point by creating replication rules.' Below is a 'Choose a template for creating replication rules' section. Two options are shown: 'Replicate objects from one or more source buckets to one or more destination buckets' (radio button not selected) and 'Replicate objects among all specified buckets' (radio button selected). An icon shows two buckets with arrows pointing between them. A warning box states: 'Creating replication rules using this template will overwrite existing replication rules for the specified buckets. If you prefer to not overwrite existing replication rules, use the other template for creating replication rules and specify one source bucket and one destination bucket.' At the bottom, a 'Buckets' section is visible with a note about enabling Bucket Versioning.

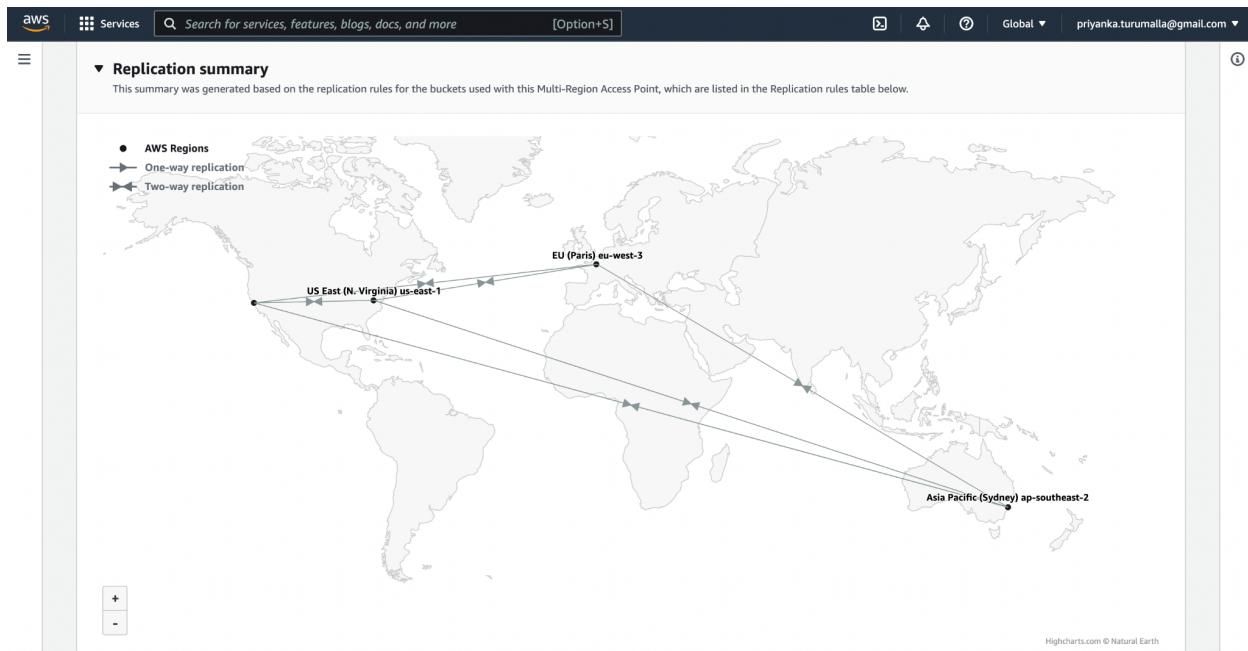
**As there were four s3 buckets, I got 12 replication rules for the same.**



The screenshot shows the 'Replication rules' table in the AWS S3 console. The top navigation bar includes the AWS logo, services menu, search bar, and user information. A green header bar indicates 'Successfully created 12 replication rules'. The table has columns for Name, Source bucket, Destination bucket, Status, and Error. All 12 rules listed are marked as 'Succeeded' with a green checkmark. The table rows show various combinations of source and destination buckets, such as 'Multi-Region Access Point : s3ptbucket3' replicating to 's3ptbucket1', 's3ptbucket2', 's3ptbucket4', and 's3ptbucket3'.

Name	Source bucket	Destination bucket	Status	Error
Multi-Region Access Point : s3ptbucket3	s3ptbucket1	s3ptbucket3	Succeeded	-
Multi-Region Access Point : s3ptbucket2	s3ptbucket1	s3ptbucket2	Succeeded	-
Multi-Region Access Point : s3ptbucket4	s3ptbucket1	s3ptbucket4	Succeeded	-
Multi-Region Access Point : s3ptbucket3	s3ptbucket2	s3ptbucket3	Succeeded	-
Multi-Region Access Point : s3ptbucket1	s3ptbucket2	s3ptbucket1	Succeeded	-
Multi-Region Access Point : s3ptbucket4	s3ptbucket2	s3ptbucket4	Succeeded	-
Multi-Region Access Point : s3ptbucket2	s3ptbucket3	s3ptbucket2	Succeeded	-
Multi-Region Access Point : s3ptbucket1	s3ptbucket3	s3ptbucket1	Succeeded	-
Multi-Region Access Point : s3ptbucket4	s3ptbucket3	s3ptbucket4	Succeeded	-
Multi-Region Access Point : s3ptbucket3	s3ptbucket4	s3ptbucket3	Succeeded	-
Multi-Region Access Point : s3ptbucket2	s3ptbucket4	s3ptbucket2	Succeeded	-
Multi-Region Access Point : s3ptbucket1	s3ptbucket4	s3ptbucket1	Succeeded	-

## Replication summary :



e) Created Lifecycle rules for first bucket I have created - s3ptbucket1 and haven chosen the lifecycle action to move the current version of objects between storage classes and expire current version of objects.

The figure is a screenshot of the AWS Lifecycle rule configuration screen. It shows a list of actions:

- Move current versions of objects between storage classes
- Move noncurrent versions of objects between storage classes
- Expire current versions of objects
- Permanently delete noncurrent versions of objects
- Delete expired object delete markers or incomplete multipart uploads

A note below the list states: "These actions are not supported when filtering by object tags or object size."

Below the actions, there is a section titled "Transition current versions of objects between storage classes". It contains two sets of dropdown menus and input fields:

- Choose storage class transitions:
  - Standard-IA → 75 → Remove
  - Glacier Deep Archive → 365 → Remove
- Add transition

At the bottom of the screen, there is a warning message with an exclamation mark icon:

**⚠ Transitioning small objects to Glacier Flexible Retrieval (formerly Glacier) or Glacier Deep Archive will incur a per object cost**

You will be charged for each object you transition to S3 Glacier Flexible Retrieval (formerly Glacier) or S3 Glacier Deep Archive. A fixed amount of storage is also added to each object to accommodate metadata for managing the object which increases storage costs. You can reduce these costs by limiting the number of objects to transition (by prefix, tag, or version), or by aggregating objects before transitioning them.

Learn more about [Glacier Flexible Retrieval \(formerly Glacier\) cost considerations](#) or review the table on Requests and data retrievals tab on the [Amazon S3 pricing page](#).

I acknowledge that this lifecycle rule will incur a one-time lifecycle request cost per object if it transitions small objects.

**Based on the given requirement Standard IA is chosen 75 days and after 75 days to make the content available for one year, objects are moved to glacier Deep Archive and the contents have to be archived for another year , no of days to expire current version is given as 730.**

The screenshot shows the AWS S3 console with the 'Transition and Expiration' tab selected. A search bar at the top has 'Days after object creation' and '730' entered. The main area is titled 'Review transition and expiration actions'. It contains two columns: 'Current version actions' and 'Noncurrent versions actions'. The 'Current version actions' column lists actions for Day 0 (Objects uploaded), Day 75 (Objects move to Standard-IA), Day 365 (Objects move to Glacier Deep Archive), and Day 730 (Objects expire). The 'Noncurrent versions actions' column shows 'No actions defined.' At the bottom right are 'Cancel' and 'Save' buttons.

**f) Creation of users:**

**Created two users as shown below :**

**Identity and Access Management (IAM) -> Users ->Add users**

- 1) User1 ( Full access - to read and write)**
- 2) User2 (No access)**

AWS Services Search for services, features, blogs, docs, and more [Option+S]

IAM > Users

Managing human user access account by account? There's a better way. Dismiss Go to Identity Center

Streamline human access to AWS and cloud apps when you enable Identity Center.

Learn more Watch how it works

One-time set up for workforce user access      Centrally manage access to multiple AWS accounts      Provide access centrally to the cloud applications your workforce uses      All with one-click access through a simple web portal

**Users (2) Info**

An IAM user is an identity with long-term credentials that is used to interact with AWS in an account.

Add users

User name	Groups	Last activity	MFA	Password a...	Active key age
user1	None	Yesterday	None	Yesterday	Yesterday
user2	None	Yesterday	None	Yesterday	Yesterday

## User 1 - Having AmazonS3FullAccess :

The screenshot shows the AWS Identity and Access Management (IAM) service interface. On the left, the navigation pane is visible with options like Dashboard, Access management, and Users. Under Users, 'user1' is selected. The main content area is titled 'Summary' for user1. It displays the User ARN (arn:aws:iam::290041667495:user/user1), Path (/), and Creation time (2022-09-21 15:45 PDT). Below this, there are tabs for Permissions, Groups, Tags, Security credentials, and Access Advisor. The 'Permissions' tab is active, showing a single policy applied: 'AmazonS3FullAccess', which is an AWS managed policy. There are buttons for 'Add permissions' and 'Add inline policy'. A note at the bottom states: 'You can generate a new policy based on the access activity for this user, then customize, create, and attach it to this role. AWS uses your CloudTrail events to identify the services and actions used and generate a policy. Learn more'.

## User2 : With no permissions

The screenshot shows the AWS IAM service interface, similar to the previous one but for user2. The navigation pane and user selection are identical. The main content area is titled 'Summary' for user2. It displays the User ARN (arn:aws:iam::290041667495:user/user2), Path (/), and Creation time (2022-09-21 15:50 PDT). The 'Permissions' tab is active, showing a message: 'Get started with permissions' with the subtext 'This user doesn't have any permissions yet. Get started by adding the user to a group, copying permissions from another user, or attaching a policy directly. Learn more'. There is a 'Add permissions' button and a 'Add inline policy' button. Below this, there are sections for 'Permissions boundary (not set)' and 'Generate policy based on CloudTrail events'.

**With the help of the console link and password for user1 tried to login and uploaded 35MB pdf file to s3ptbucket1.**

The screenshot shows the AWS S3 'Upload: status' page. At the top, a green banner indicates 'Upload succeeded'. Below it, the 'Summary' section shows the destination 's3://s3ptbucket1' and the upload status: 'Succeeded' (1 file, 35.2 MB (100.00%)) and 'Failed' (0 files, 0 B (0%)). On the left, tabs for 'Files and folders' (selected) and 'Configuration' are visible. The main area displays a table titled 'Files and folders (1 Total, 35.2 MB)' with one item: 'Mastering Cloud Computing.pdf' (application/pdf, 35.2 MB, Status: Succeeded). The right sidebar includes links for Account, Organization, Service Quotas, Billing Dashboard, Security credentials, and Settings, along with 'Switch role' and 'Sign out' buttons.

**Whereas for user2 since permissions are not created , buckets are not visible**

The screenshot shows the AWS S3 'Amazon S3' dashboard. The left sidebar has sections for 'Buckets' (selected), 'Access Points', 'Object Lambda Access Points', 'Multi-Region Access Points', 'Batch Operations', 'Access analyzer for S3', 'Block Public Access settings for this account', 'Storage Lens' (with 'Dashboards' and 'AWS Organizations settings'), 'Feature spotlight' (with a blue notification badge), and 'AWS Marketplace for S3'. The main content area shows an 'Account snapshot' with a 'View Storage Lens dashboard' button. Below it is a 'Buckets' table with columns: Name, AWS Region, Access, and Creation date. The table shows 'No buckets' and 'No buckets' under the respective columns. A 'Create bucket' button is located at the bottom of the table.

**Likewise, permission denied error has been shown to access the dashboard as well.**

The screenshot shows the AWS Storage Lens dashboard for a specific account. At the top, there is a navigation bar with the AWS logo, a search bar, and user information. Below the navigation bar, the dashboard title is "default-account-dashboard". There are several tabs at the top: "Overview", "Account", "AWS Region", "Storage class", and "Bucket". A prominent red-bordered box contains an error message: "You don't have permissions to view this Storage Lens dashboard. You or your AWS administrator must update your IAM permissions to allow s3:GetStorageLensConfiguration and s3:GetStorageLensDashboard permissions. After you obtain the necessary permission, choose Refresh. Learn more about Identity and access management in Amazon S3." Below this message, there is a link "▶ API response".

**g) To block the access from 'Cuba'. Choose cloudfront -> Distributions-> Edit the geographic restrictions.**

The screenshot shows the "Edit geographic restrictions" page for a CloudFront distribution. The top navigation bar includes the AWS logo, search bar, and user information. The page title is "Edit geographic restrictions". A sidebar on the left shows "Settings" and "Info". Under "Settings", the "Restriction type" is set to "Block list". In the "Countries" section, "Cuba" is listed in a dropdown menu. At the bottom of the page are "Cancel" and "Save changes" buttons.

**Architecture Diagram:**

