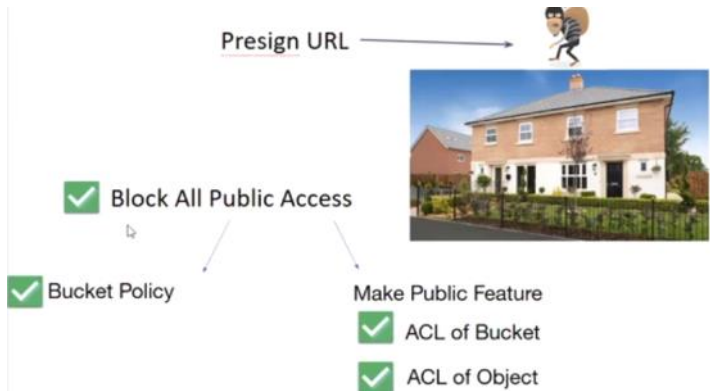


## AWS S3-2



Atic ya da dynamic olan hesabima attigim objeleri insanların disardan gormesini istiyorum. Oyleyse girecek kisilerin bu image ve videolara erisimi olmasi lazim. Bunun yontemi de oncelikli olarak block all public accessi kaldirmamiz. Sonrasinda ise ister bucket policy ile ister Access Control list(ACL) ile public yapip yapmama kararini vermis oluyorum. presign URL ise objeyi kısa aralik icin public olmamasina ragmen publicmis gib aciyor.



### Static Website Hosting



S3 icinde hem ec2 hem kodu var.

Amazon S3'ü kullanarak, NGINX gibi bir sunucuya ihtiyaç duymadan statik web sitelerini (HTML, CSS, JavaScript dosyaları) barındırabilirsiniz. Yani, dosyalarınızı S3'e yükleyerek sitenizi kolayca yayına alabilirsiniz.

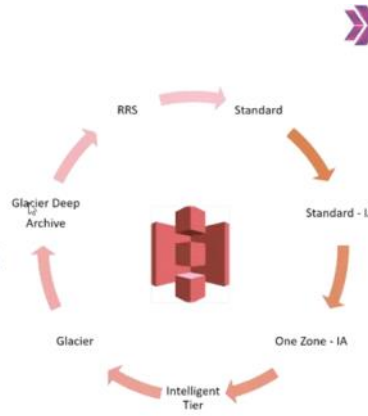
## 1

## Lifecycle Management

## Lifecycle Management

### What is Lifecycle Management?

- Lifecycle management is to determine how an object will be subject to a **storage policy** during the time it is stored in S3.
- So, it is used to ;
  - Transition objects to another storage class,
  - Archive objects,
  - Delete objects after a specified period of time.

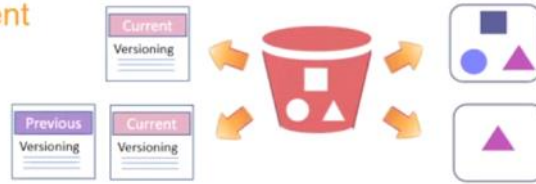


Lifecycle Management, Amazon S3'te saklanan dosyaların ne kadar süreyle hangi depolama politikasına tabi olacağını belirlemektir. Bu yönetim şunlar için kullanılır:

1. Dosyaları farklı depolama sınıflarına taşımak.
2. Dosyaları arşivlemek.
3. Belirli bir süre sonra dosyaları silmek.
4. Intelligent tier gibi değerlendirme yapıp dosyayı bi uste ya da ata tasıamaz. Burada tam bir detay veririz bir ay sonra sonra mesela alta al deriz ve alır.

## Lifecycle Management

### Lifecycle Management



Object base degil bucket base. Bucketlarımıza life cycle manegementleri yukluyoruz ve o kurallara gore bucket isiliyor.

Lifecycle kurallari belirlerken taglere ya da prefixlere gore kurallari daraltabiliyoruz filtreleyebiliyoruz.

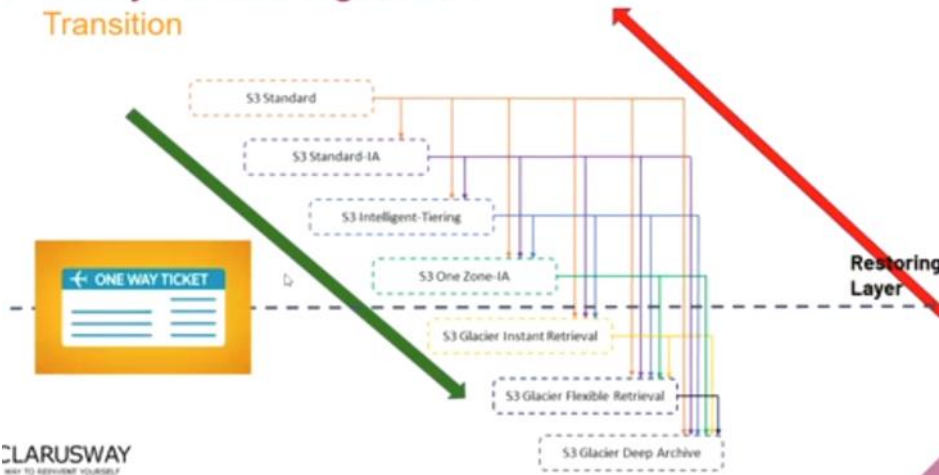
Versiyonlama acik.

Aws belirledigi minimum sure kadar beklemesi lazim transition icin.

- A lifecycle policy is **bucket-base component**.
- You can narrow Lifecycle Management by using a **prefix** or **tag**
- Available for both **current** and **previous**
- Minimum days for transition

## Lifecycle Management

### Transition



Transition islemi maddi anlamda bizi rahatlatmasi icin yapilan bir durum.

Ben bucketimdakilerin ne kadar sure bana lazim olacagini biliyorum ve buna gore bir manegement olusturuyorum asagiya dogru bir traansition belirliyorum. Zaten asagiya dogru inebilirz ama yukari cikamayiz.

Normalde her class bir altindakine gecis yapabiliyorken one zonne IA glacier Instant Retrievala gecis yapamiyor. Bu bir kural

## ► Lifecycle Management

### Advantage of Lifecycle Management



Expression sayesinde istenmeyen objeleri silebiliyorsunuz. Objelerin kullanım sıklığına göre alt classa taşıma olayı ile money save yapılabiliniyor. Manuel olsaydı erisim mevzularındaki hassasiyetler gözden kaçabilirdi. Life cycle management sayesinde bu ayrıntılar gözden kaçmamış oluyor. Aynı zamanda bucketi düzenlemiş oluyor.

- It provides you to **arrange your S3 inventory**.
- You can **save money** by transition the objects to the cost-effective storage class,
- You can **get rid of redundant objects**.

## ► Lifecycle Management

### Enabling Lifecycle Management

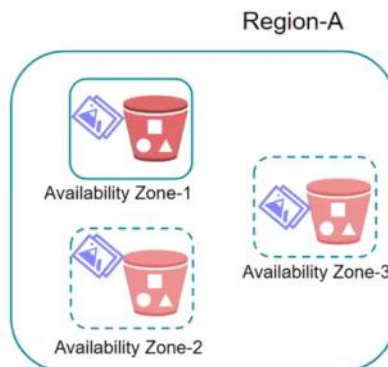
#### Lifecycle Management Path

S3>Bucket> **Management** > Lifecycle>Add a Lifecycle Rule

## 2 Bucket Replication

## ► Bucket Replication

### What is Bucket Replication?



S3 bucketin manegement tabinde ilerliyor. Normalde min 3 AZ seciyorduk ve bucketimizdaki objeleri min 3 AZ de tutuyordu s3 bucketlerini ama biz bu 3 AZ ye de erisemiyorduk. Bu AWS in kendinden kaynakli bir sorun cikma ihtimaline karsi aldigi onlemdi.

CLARUSWAY

## ► Bucket Replication

### What is Bucket Replication?



Biz bu minimum 3 Azye erisemedigimiz icin bunu biz kendimiz kendi belirledigimiz objelerce baska bir bucketta yine saklaniyor olacak bu yontemle. Bunun icin 2 yontem var: Cross region ve Same region replication. Crossta bunu baska bir regiona yedeklemesini yapar olabilecek ihtimallere karsi. Burada ucretlendirmeye tabi tutulur farkli region olmasi sebebiyle. Bir de musterilerin bulundu gu yerlere gore bir region secersek latency noktasinda kolaylik saglamis oluruz. Samede ise ayni regionda bunu gerceklestirir. Burada ayni region olmasindan dolayi ekstra ucretlendirme olamaz.

- Replication is to **copy** the objects in the buckets to **another bucket** in **different region** or in the **same region**.
- There are **2 types** of replication in S3:
  - **Cross-Region Replication (CRR)** is used to copy objects in different AWS Regions
  - **Same-Region Replication (SRR)** is used to copy objects in the same AWS Region

## ► Bucket Replication

### Why Replication?

- **Uyumluluk gereksinimleri:** Yasal ve duzenleyici gerekliliklere uyum saglamak.
- **Yedekleme:** Verilerin yedeklenmesi icin kopyalarni olusturmak.
- **Gecikmeyi azaltma:** Verilere daha hizi erisim saglamak icin veriyi farkli bolgelerde tutmak.
- **Ilgili verileri tek bir kovada toplamak:** Ilgili verileri bir araya getirecek yonetimi kolaylastirmak.

- Compliance requirements
- Backup
- Minimize latency
- Aggregating related data into a single bucket.

## ► Bucket Replication

### Replication Features

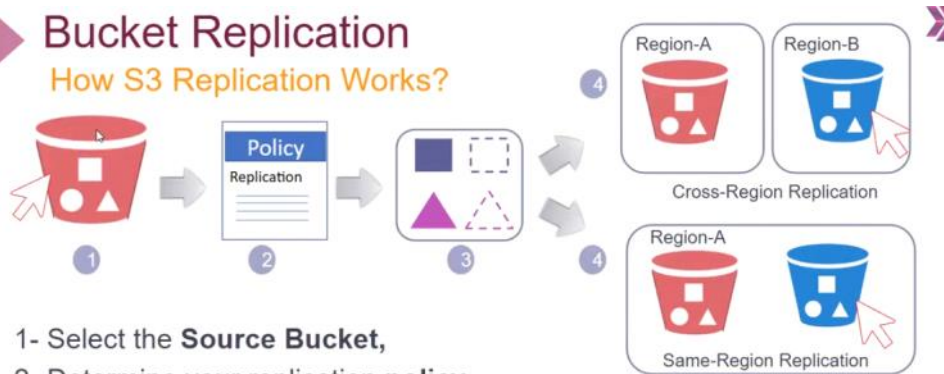


Replike ederken benim belirledigim storage classta replike edilsin diyebiliriz. Aninda replikesi yapiliyor kucuk dosyalarda ama buyk dosyalar icinde 15 dk icinde aliniyor replikesi  
Replike ederken filtreleyerek isteidigimiz kriterlerce replike edebiliriz.

- Keep the object's **metadata** while replicating
- **Replicating in Different storage class**
- Option of changing ownership **of the** replicated object
- Synchronizing **within 15 minutes**.
- Narrowing replication based on **prefix or tag**

## ► Bucket Replication

### How S3 Replication Works?



- 1- Select the **Source Bucket**,
- 2- Determine your replication **policy**,
- 3- Select your data set by object **tag, prefix or entire bucket**,
- 4- Select the **Destination Bucket** in the same region or in different region.