

An Overview of IBM Cloud Object Storage

James Belton
October 2018

Introduction

Object Storage is a ubiquitous element of storage in the cloud and the IBM Cloud is no exception. It offers both reliable and cheap storage for files but there are lots of different options to consider and lots to know in order to get the best from it.

I am an IBM Cloud Technical Subject Matter Expert, working in the UK and over the past couple of years, I've gotten to know IBM Cloud Object Storage (COS) pretty well, having created a few 'how-to' guides and videos on the subject.

I've created this document as a higher-level guide to help users to better understand what IBM Cloud Object Storage is and the various options that are available to help them get going. It's not exhaustive and isn't intended to be a substitute for the official documentation (which can be found at <https://console.bluemix.net/docs/services/cloud-object-storage/about-cos.html#about-ibm-cloud-object-storage>) but hopefully it provides a decent 'getting started' guide to help build knowledge for first time or novice users.

Comments and feedback are welcome – if there is something that you'd like to see covered here or covered in more detail, let me know and I'll look to incorporate it into a later version.

Cheers,

James

Table of Contents

Introduction	2
What is IBM Cloud Object Storage and What is it Used For?	4
What is an Object?	4
What wouldn't you store there?	4
How is Data Stored?	4
What Are the Different Flavours of Bucket?	4
Bucket Resiliency	4
Cross Region Buckets	5
Regional Buckets	5
Single Site Buckets	5
Bucket Storage Classes	5
Standard	5
Vault	6
Cold Vault	6
Flex	6
IBM Cloud Object Storage Archive	6
Placing objects into the Archive	6
Policies to move data between other types of bucket	7
Retrieving Data from the Archive	7
Getting Objects Into and Out of Cloud Object Storage	7
Via the IBM Cloud Console	7
Via the COS API	7
Via the S3 CLI or S3 Compatible Tools	8
Providing Others Access to Cloud Object Storage	8
Related Services	8

What is IBM Cloud Object Storage and What is it Used For?

IBM Cloud Object Storage (COS) is pretty much what it says it is: an offering on the IBM Cloud which provides storage for objects. The storage can be accessed by users or applications to add or retrieve objects, and this is done either programmatically through the COS API, or via the COS GUI available through the IBM Cloud Console. COS also supports S3 API commands, meaning that users of the AWS S3 CLI and other third-party tools that use S3 commands can access and use COS.

What is an Object?

An object is pretty much any file that you would store in a traditional file system. This includes things like text files, image files, audio files, office files, pdf's and so on. You can store files of any size, up to 10TB. COS is in theory 'unlimited' meaning you can store as much data as you like there, paid for based on how much storage is used.

What wouldn't you store there?

COS is not suitable for storing files that run operating systems or databases. For example, you would not install Linux or Windows on COS and be able to run a server from it, nor would you place database files onto COS. However, COS is a good place to store backups of machine images or databases.

How is Data Stored?

In Object Storage, data is stored in 'Buckets' and there is a limit of 100 buckets to an account. A bucket is effectively a means of organizing or collecting data together.

In COS, bucket names must be unique as since they share a single namespace across the globe.

When uploading data to COS, it goes into a bucket which you define. Think of a bucket as similar to a 'top level' folder in a traditional file storage system. You cannot nest buckets inside buckets – if you need to have some kind of hierarchy within the bucket, then this can be achieved with prefixes.

What Are the Different Flavours of Bucket?

When creating a bucket, there are a number of choices. These can be split into:

- Resiliency
- Storage Class

Bucket Resiliency

Resiliency refers to how tolerant the bucket is to failure and therefore how available the data in that bucket is. There are three choices available for resiliency when creating a bucket and the choice made reflects the importance placed on the availability of the objects in the bucket. The different resiliency levels are:

- Cross Region
- Regional

- Single Site

Once a bucket has been created, you cannot change its resiliency. If a change is needed, you must create a new bucket and transfer objects into it. You may delete buckets that are no longer needed, though they must be empty first.

Cross Region Buckets

Cross Region buckets store their objects across three regions within a geographical area. For example, the 'us-geo' means that the data will be stored in multiple sites in the regions us-south and us-east. The data can then be retrieved using any one of those copies. This option is most resilient to failure, since three entire regions would need to fail. Storing objects in Cross Region buckets is more expensive than Regional or Single Site buckets.

Regional Buckets

Regional Buckets store their objects in multiple data centre sites within a single region. For example, the 'eu-gb' region means that the data will be stored at several sites within the UK. Data can be retrieved using any one of those copies. This option is more resilient to failure than Single Site buckets because data is stored at more than one site but less resilient than Cross Region buckets, since data is only stored in one region. In the unlikely event that a region fails, the data will be unobtainable. Storing data in Regional Buckets is more expensive than Single Site buckets but less expensive than Cross Region Buckets. This offering may be suitable for applications where data sovereignty is a constraint.

Single Site Buckets

Single Site Buckets store their objects at a single site. For example, the 'ams03' site means that the data will be stored at the site called 'ams03' (Amsterdam 03). If the site fails, the data will not be available for use. This is the least resilient offering but also the least expensive. This offering is suitable where objects do not need to be highly available (e.g. test data) or where there are stricter data sovereignty considerations.

Bucket Storage Classes

There are four different storage classes available in COS, which align with different expectations of frequency of access which allow users to balance access needs against cost. The four different Storage Classes are:

- Standard
- Vault
- Cold Vault
- Flex

Once a bucket has been created, you cannot change its storage class. If a change in storage class is needed, you must create a new bucket and transfer objects into it.

Standard

Standard buckets are suitable for objects which are going to be accessed frequently – meaning multiple times per month. These are also referred to as 'active workloads'. There are no fees for data retrieval.

Vault

Vault buckets are for what are referred to as 'cool workloads' – in other words objects which are accessed less frequently than monthly but are still retrieved more frequently than objects which might be better suited to Cold Vault. For Vault, there is a retrieval charge. Also, for billing, there is a minimum object size of 128 KB and a minimum storage period of 30 days (referred to as 'a threshold for object size and storage period' in the documentation). This means that objects stored that are less than 128 KB in a Vault bucket will be charged as though they are 128 KB in size and that if you place an object into a Vault bucket and delete it within 30 days, 30 days storage will still be charged.

Cold Vault

Cold Vault buckets are used for objects which are considered to be 'cold' or effectively archived. This means that they are accessed very infrequently, typically less than every 90 days. A retrieval fee is levied for reads of these objects, which is higher than the retrieval charge made for Vault storage. Similar to Vault, there are also minimum billing thresholds for object size and storage period, though these are 256 KB and 90 days respectively.

Flex

There are occasions where it is more difficult to predict the access patterns for data and so, Flex buckets offer a solution. With Flex, there is an access charge but if the costs of storage and retrieval exceed a cap, then retrieval costs are not billed. If the data isn't accessed frequently, then Flex buckets work out to be more cost effective than Standard buckets. Conversely, if the 'cool' data becomes accessed more frequently than expected, it becomes more cost effective than Vault or Cold Vault.

IBM Cloud Object Storage Archive

There is a further COS tier known as 'Archive' and this is the lowest cost option for data that is very rarely accessed. The Archive tier is comprised of off-line storage and is intended for long-term storage of objects, such as those which need to be kept for regulatory compliance purposes. There is a higher retrieval charge for objects stored in the Archive tier and retrieval itself may take up to 15 hours.

Placing objects into the Archive

When creating a bucket, you can set up an optional Archive policy. What this means is that when an object has been in the bucket for the time period set by the archive policy, it will be moved to an Archive. Time periods are Days, Weeks, Month and Years.

You may also set up an Archive policy on an existing bucket, but the policy will only apply to objects that are subsequently added. For example, setting an Archive Policy on an existing bucket of 1 day, will mean that all new objects placed in the bucket after the policy is activated will be moved to the Archive but any objects that existed in the bucket when the policy was activated will not.

If you want to immediately archive all new objects in bucket, set the 'time to archive' to 0 days. This would be a way to archive existing objects – to create a new bucket with an archive policy of 0 days and to then move those existing objects to the new bucket. They will then be instantly archived.

Note that when using an archive policy of less than 30 or 90 days on Vault or Cold Vault buckets respectively, you will still be charged for 30 or 90 days storage in those buckets, as well as for storage in the Archive.

Policies to move data between other types of bucket

You cannot currently set policies which automatically move objects between other types of bucket. For example, there is no way to set a policy which automatically moves objects from a Standard bucket to a Vault bucket or from a Vault bucket to a Cold Vault bucket. This may however be possible programmatically via the API or through 3rd party tools.

Retrieving Data from the Archive

When you retrieve data from an archive, it will be restored to a bucket of the same storage class as it originally resided in (at the time it was archived). Retrieval can take up to 15 hours to complete and a charge will be made for it.

You can also set a limit for the number of days that a restored object is available, after which the restored copy be deleted.

Getting Objects Into and Out of Cloud Object Storage

There are three main ways to achieve this:

- Via the console in IBM Cloud
- Via the COS API
- Via the AWS CLI or S3 compatible tools

Via the IBM Cloud Console

You can achieve a lot with COS through the IBM Cloud Console. As well as creating buckets you can use the console as an easy means to upload and downloads objects, though note that there is a 200 MB size limit on uploads too. However, using the console is not practical in most situations – either because of the access privileges you need to give to allow other users to do so or because programs cannot make use of it.

Via the COS API

The COS API provides programmatic and command-line access to COS. The API is fully documented and provides a means to achieve any function within COS and can be called by a wide variety of programming languages. The full documentation is available at <https://console.bluemix.net/docs/services/cloud-object-storage/api-reference/about-api.html> .

Via the S3 CLI or S3 Compatible Tools

Since COS also supports S3, which is used by AWS S3 (the Amazon cloud implementation of object storage), it can be used via the AWS CLI as well as with tools that support S3, such as Cyberduck, Transmit and Cloudberry. Cyberduck and Transmit are tools that support file transfers between user machines and COS, while Cloudberry is a backup tool which can send backup files directly to COS.

You can read a guide which shows the process for connecting Transmit to COS at <https://github.com/jamesbeltonIBM/Connecting-an-s3-API-Compatible-Third-Party-Utility-to-work-with-IBM-Cloud-Object-Storage->

Providing Others Access to Cloud Object Storage

COS uses IBM Cloud Identity and Access Management (IAM) as its main method of providing access. If using S3 commands to work with COS, then 'HMAC' access needs to be configured to provide the right kinds of access keys.

Using IAM, a user or service (application) can be granted granular access to COS, down to the bucket level. It's also possible to make single objects in COS publicly available via URLs using either IAM or S3.

You can read more about IBM Cloud Identity and Access Management at <https://github.com/jamesbeltonIBM/IBMCloudIAM>

You can read more about making objects in COS publicly available at <https://github.com/jamesbeltonIBM/Making-Objects-Publicly-Available-from-IBM-Cloud-Object-Storage->

Related Services

If you're interested in using the SQL Query service to use SELECT statement to query csv, JSON, ORC or Parquet files stored in Cloud Object Storage, take a look at a blog post I wrote here: <https://thebluemixdigest.wordpress.com/2018/10/25/ibm-cloud-sql-query/>