CLOUD COMPUTING FRAMEWORK

Open-source cloud frameworks: A work in progress

Nimble and fast, open-source frameworks can simplify application deployment in the cloud. But they're not for everyone

When IT consultancy OpenCredo set out to launch three new applications within seven months for a major insurance underwriter, it had three goals in mind: Trim development time from the usual years-long pace, allow for frequent changes from the client, and build a system that can handle unpredictable traffic spikes.

By using the Cloud Foundry open-source framework along with other open-source software, OpenCredo eliminated "heavy lifting" such as configuring virtual machines and adjusting the size of storage volumes, says CEO Russell Miles. The framework allowed developers to write code locally, share it with the client, and automate the integration, testing, and deployment of application components.

Among other advantages, Cloud Foundry makes it easier to scale an application by adding more instances without downtime, Miles says. Because of the way it works with other open-source software, new features can be added in minutes rather than hours.

Even with all those benefits, open-source cloud frameworks like Cloud Foundry are a work in progress. Many manage only physical servers or stand-alone applications, leading customers who need more sophisticated capabilities to create their own frameworks. However, they offer compelling value because they mask the complexity of cloud computing setups, and the open-source model is an attractive way to do that.

Understanding the Basics

The term "framework" is used to loosely describe collections of anything from development tools to middleware to database services that ease the creation, deployment and management of cloud applications. Those that work at the level of servers, storage and networks are infrastructure-as-a-service (IaaS) frameworks. Those that operate at the higher level of applications are platform-as-a-service (PaaS) frameworks.

Among the most popular IaaS frameworks are OpenStack, Eucalyptus, and the Ubuntu Cloud infrastructure. Citrix recently announced it was making its formerly proprietary CloudStack IaaS platform part of the open-source Apache project. Gartner analyst Lydia Leong wrote in her blog that this is "big news" because CloudStack is much more stable and production-ready than the "unstable" and "buggy" OpenStack.

Popular PaaS frameworks include Heroku, Cloud Foundry (backed by VMware), and Red Hat's OpenShift, which is built on a foundation of Red Hat Enterprise Linux with support for a variety of languages and middleware through the use of "cartridges."

Customers often use multiple frameworks and associated tools. One example is the use of OpenStack to provision virtual machines, and Opscode Chef to create "recipes" describing how servers should be configured, says Opscode co-founder Jesse Robbins. The further up the "stack" a platform operates, the less work the customer must do, but they also have less control over the infrastructure components, says Matt Conway, CTO at online backup vendor Backupify.

Beyond easing cloud creation, most frameworks claim to make it easier to move cloud deployments among public and private clouds to get the lowest cost and best service. For example, Eucalyptus is meant to provide an Amazon EC2-compatible API that runs on top of Ubuntu Linux (the version of Linux underpinning the Ubuntu Cloud), "so apps authored for EC2 should be transplantable to one's own data center running Eucalyptus," says Conway. "Deltacloud was an initiative by Red Hat to create a 'cloud API' to abstract your application away from vendors like Amazon, and it would proxy your requests to the actual Amazon API."

For online storage vendor CX, OpenStack provides the flexibility to use other cloud vendors besides Amazon "if [Amazon's] services become too expensive or otherwise unsuitable," says CX CTO Jan Vandenbos.

Anthony Roby, a senior executive in Accenture's advanced systems and technology group, says the word "framework" is often misused, and offerings such as Eucalyptus or OpenStack are "not frameworks at all," but "products you can extend or use to build your own infrastructure cloud." However, most observers define frameworks as software building blocks used to create cloud-based services for users.

Three Steps

Choosing an Open-Source Framework

1. Evaluate which components of a "framework" you need, as well as the level (i.e., infrastructure, platform or both) at which you need to reduce your workload.
2. Evaluate the level of support and professional services you'll need.
3. Check support for the specific cloud providers you might want to use, and whether your platform supports the added services (such as replication) your applications may require.

Amazon Web Services - Amazon S3

Amazon S3 (Simple Storage Service) is a scalable, high-speed, low-cost web-based service designed for online backup and archiving of data and application programs. It allows to upload, store, and download any type of files up to 5 TB in size. This service allows the subscribers to access the same systems that Amazon uses to run its own web sites. The subscriber has control over the accessibility of data, i.e. privately/publicly accessible.

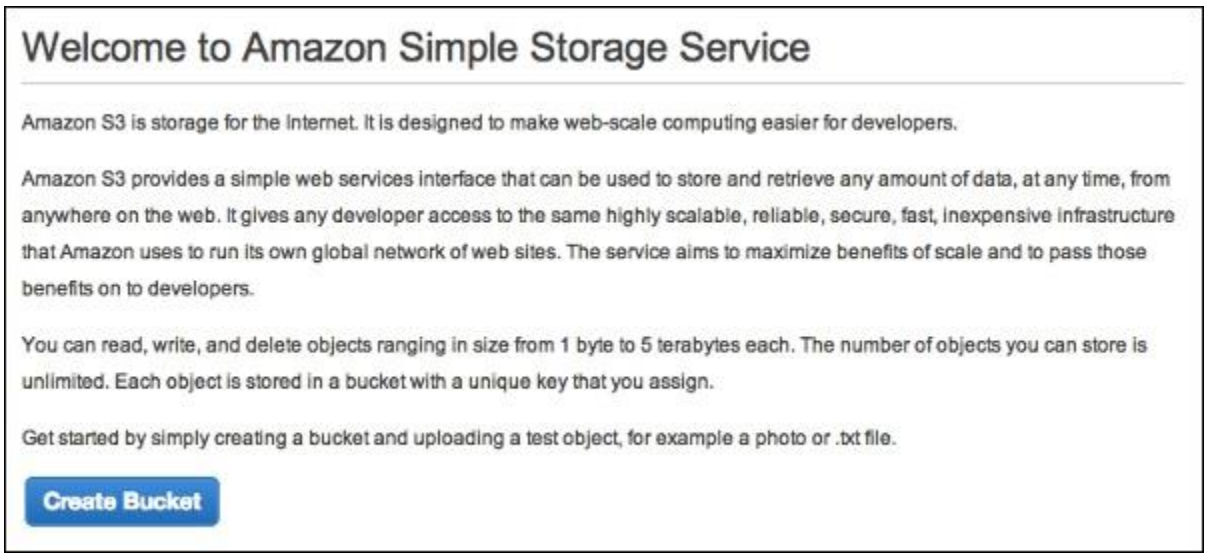
How to Configure S3?

Following are the steps to configure a S3 account.

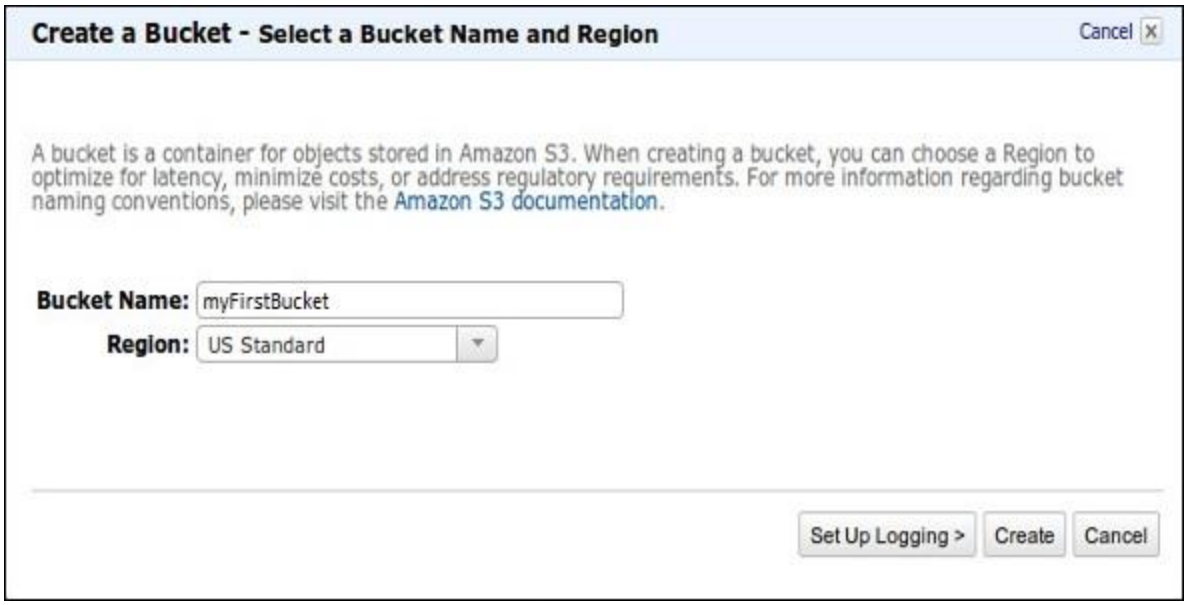
Step 1 − Open the Amazon S3 console using this link − https://console.aws.amazon.com/s3/home

Step 2 − Create a Bucket using the following steps.

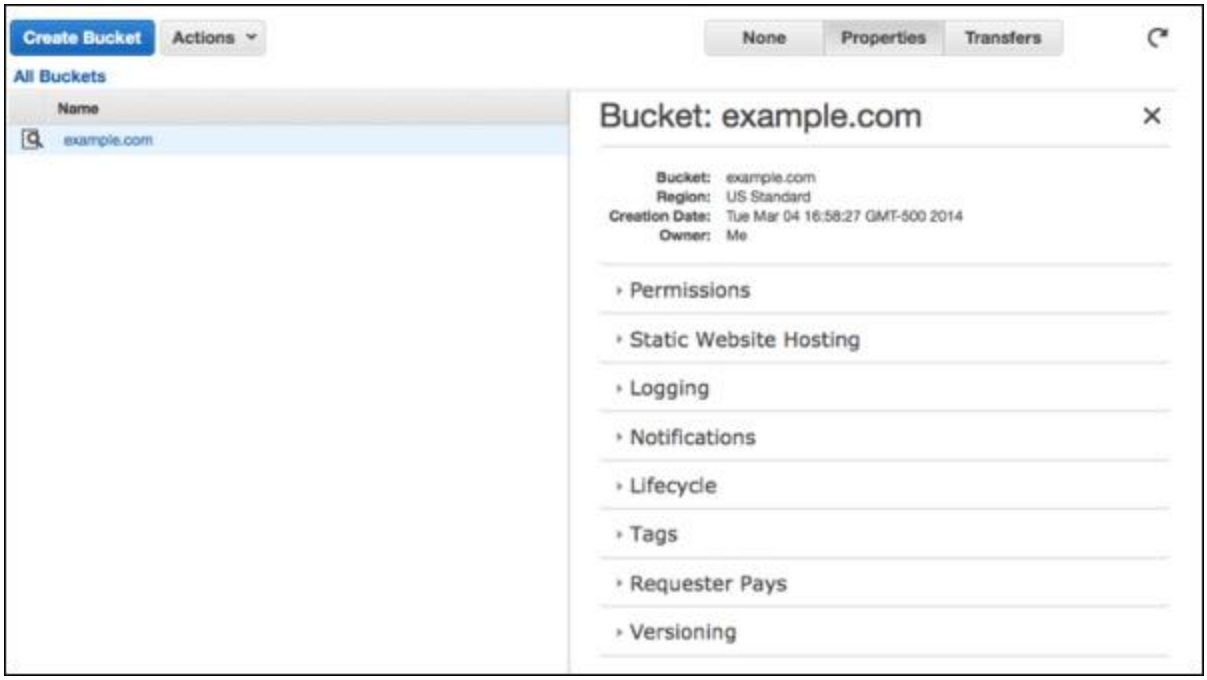
- A prompt window will open. Click the Create Bucket button at the bottom of the page.



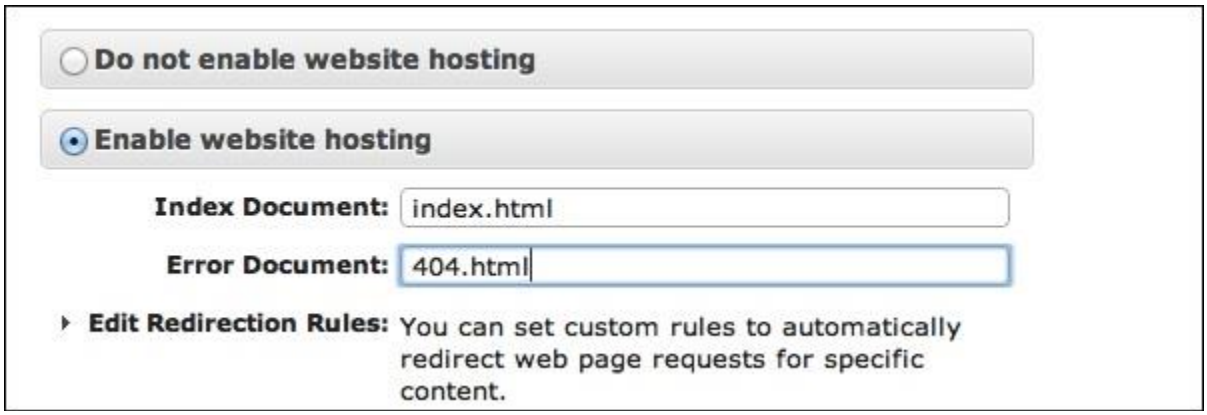
- Create a Bucket dialog box will open. Fill the required details and click the Create button.



- The bucket is created successfully in Amazon S3. The console displays the list of buckets and its properties.

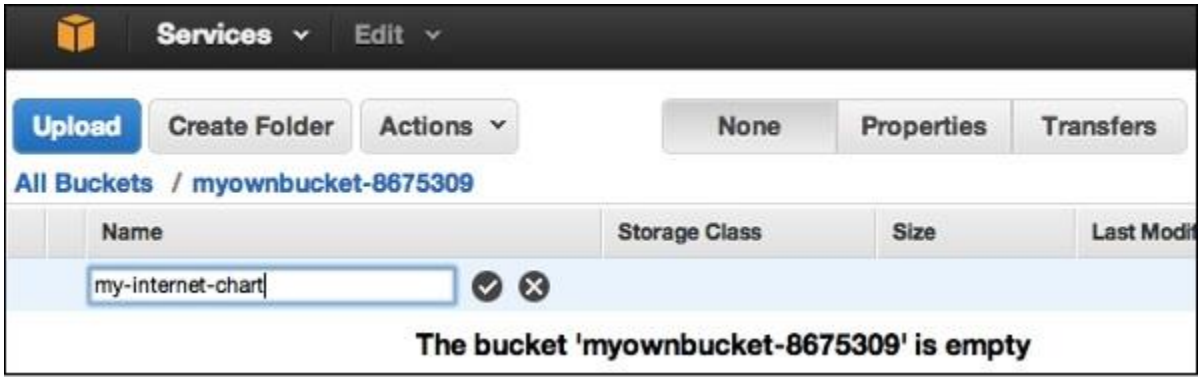


- Select the Static Website Hosting option. Click the radio button Enable website hosting and fill the required details.



Step 3 − Add an Object to a bucket using the following steps.

- Open the Amazon S3 console using the following link − https://console.aws.amazon.com/s3/home
- Click the Upload button.



- Click the Add files option. Select those files which are to be uploaded from the system and then click the Open button.

- Click the start upload button. The files will get uploaded into the bucket.

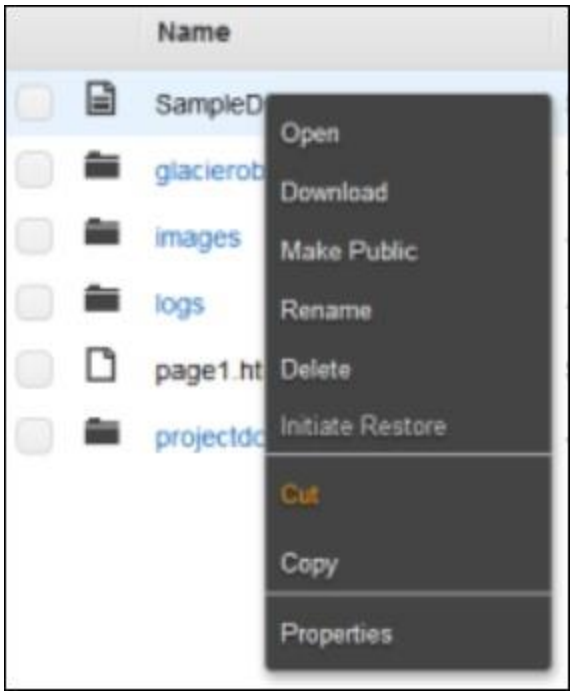
To open/download an object − In the Amazon S3 console, in the Objects & Folders list, right-click on the object to be opened/downloaded. Then, select the required object.

How to Move S3 Objects?

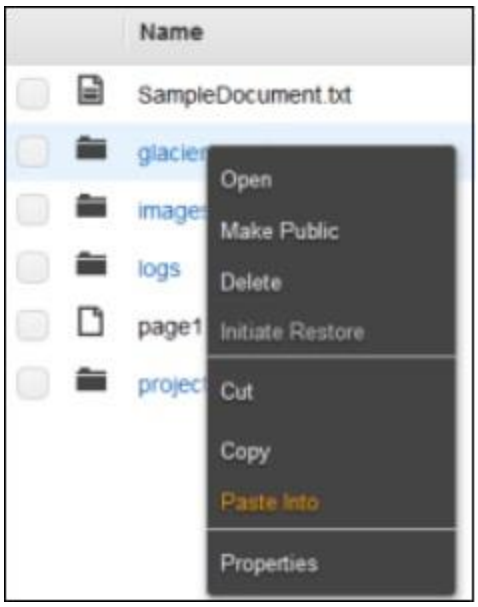
Following are the steps to move S3 objects.

step 1 − Open Amazon S3 console.

step 2 − Select the files & folders option in the panel. Right-click on the object that is to be moved and click the Cut option.



step 3 − Open the location where we want this object. Right-click on the folder/bucket where the object is to be moved and click the Paste into option.

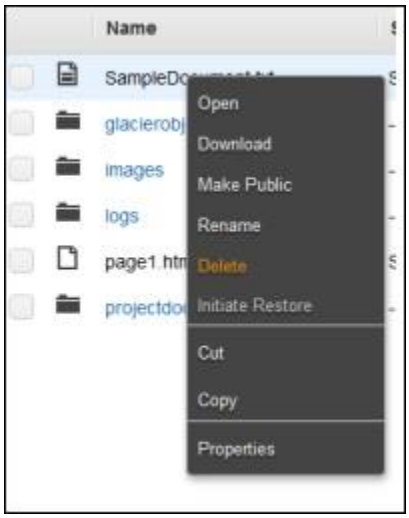


How to Delete an Object?

Step 1 − Open Amazon S3.

Step 2 − Select the files & folders option in the panel. Right-click on the object that is to be deleted. Select the delete option.

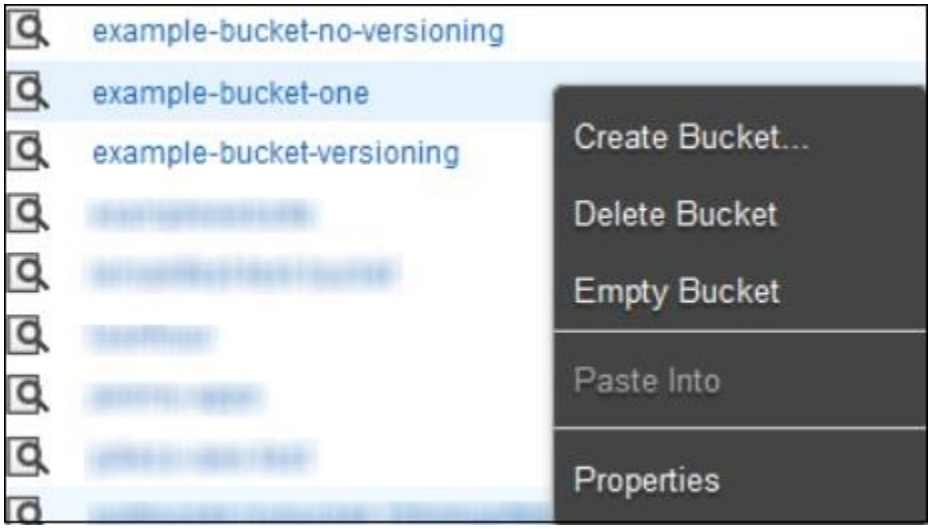
Step 3 − A pop-up window will open for confirmation. Click Ok.



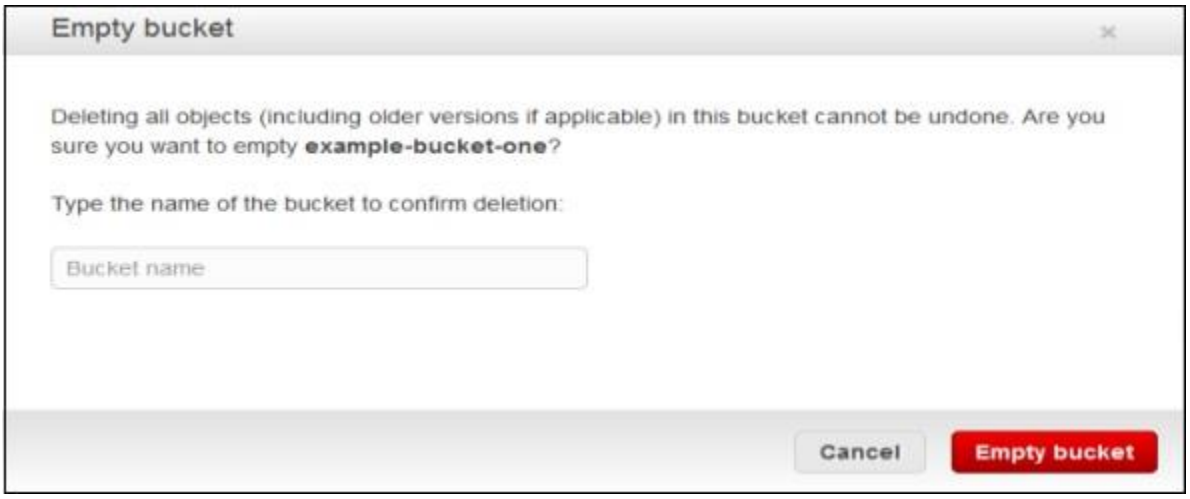
How to Empty a Bucket?

Step 1 − Open Amazon S3 console.

Step 2 − Right-click on the bucket that is to be emptied and click the empty bucket option.



Step 3 − A confirmation message will appear on the pop-up window. Read it carefully and click the **Empty bucket** button to confirm.



Amazon S3 Features

- **Low cost and Easy to Use** − Using Amazon S3, the user can store a large amount of data at very low charges.
- **Secure** − Amazon S3 supports data transfer over SSL and the data gets encrypted automatically once it is uploaded. The user has complete control over their data by configuring bucket policies using AWS IAM.
- **Scalable** − Using Amazon S3, there need not be any worry about storage concerns. We can store as much data as we have and access it anytime.
- **Higher performance** − Amazon S3 is integrated with Amazon CloudFront, that distributes content to the end users with low latency and provides high data transfer speeds without any minimum usage commitments.
- **Integrated with AWS services** − Amazon S3 integrated with AWS services include Amazon CloudFront, Amazon CLoudWatch, Amazon Kinesis, Amazon RDS, Amazon Route 53, Amazon VPC, AWS Lambda, Amazon EBS, Amazon Dynamo DB, etc.

What is AWS S3: Overview, Features and Storage Classes Explained

With more than 32 percent of the world's public cloud share, it's no surprise that Amazon Web Services (AWS) serves more than 190 countries with scalable, reliable, and low-cost cloud infrastructure. One of its most powerful and commonly used storage services is Amazon S3. S3 ("Simple Storage Service") enables users to store and retrieve any amount of data at any time or place, giving developers access to highly scalable, reliable, fast, and inexpensive data storage. Designed for 99.999999999 percent durability, S3 also provides easy management features to organize data for websites, mobile applications, backup and restore, and many other applications.

In this article, we'll talk about the following topics:

- What is cloud storage?
- Types of AWS storage
- Before AWS S3
- What is AWS S3?
- AWS S3 Benefits
- AWS Buckets and Objects
- How does Amazon S3 work?
- AWS S3 Features

What is Cloud Storage?

Cloud storage is a web service where your data can be stored, accessed, and quickly backed up by users on the internet. It is more reliable, scalable, and secure than traditional on-premises storage systems.

Cloud storage is offered in two models:

1. Pay only for what you use
2. Pay on a monthly basis

Now, let's have a look at the different types of storage services offered by AWS.

We'll eventually take an in-depth look at the S3 service. But before we get to that, let's have a look at how things were before we had the option of using Amazon S3.

Before AWS S3

Organizations had a difficult time finding, storing, and managing all of your data. Not only that, running applications, delivering content to customers, hosting high traffic websites, or backing up emails and other files required a lot of storage. Maintaining the organization's repository was also expensive and time-consuming for several reasons. Challenges included the following:

1. Having to purchase hardware and software components
2. Requiring a team of experts for maintenance
3. A lack of scalability based on your requirements
4. Data security requirements

These are the issues AWS S3 would eventually solve. So, what exactly is AWS S3?

What is AWS S3?

Amazon S3 (Simple Storage Service) provides object storage, which is built for storing and recovering any amount of information or data from anywhere over the internet. It provides this storage through a web services interface. While designed for developers for easier web-scale computing, it provides 99.999999999 percent durability and 99.99 percent availability of objects. It can also store computer files up to 5 terabytes in size.

AWS S3 Benefits

Some of the benefits of AWS S3 are:

- Durability
- Low cost
- Scalability
- Availability
- Security

- Flexibility
- Simple data transfer

Let’s have a look at some of the major components of the AWS S3 storage service.

AWS Buckets and Objects

An object consists of data, key (assigned name), and metadata. A bucket is used to store objects. When data is added to a bucket, Amazon S3 creates a unique version ID and allocates it to the object.

Like we saw in the example above, first off, a user creates a bucket. When this bucket is created, the user will specify the region in which the bucket is deployed. Later, when files are uploaded to the bucket, the user will determine the type of S3 storage class to be used for those specific objects. After this, users can define features to the bucket, such as bucket policy, lifecycle policies, versioning control, etc.

Now, let’s talk about the different storage classes offered by Amazon S3.

Amazon S3 Storage Classes

Let’s have a look at the different storage classes using the example of a school:

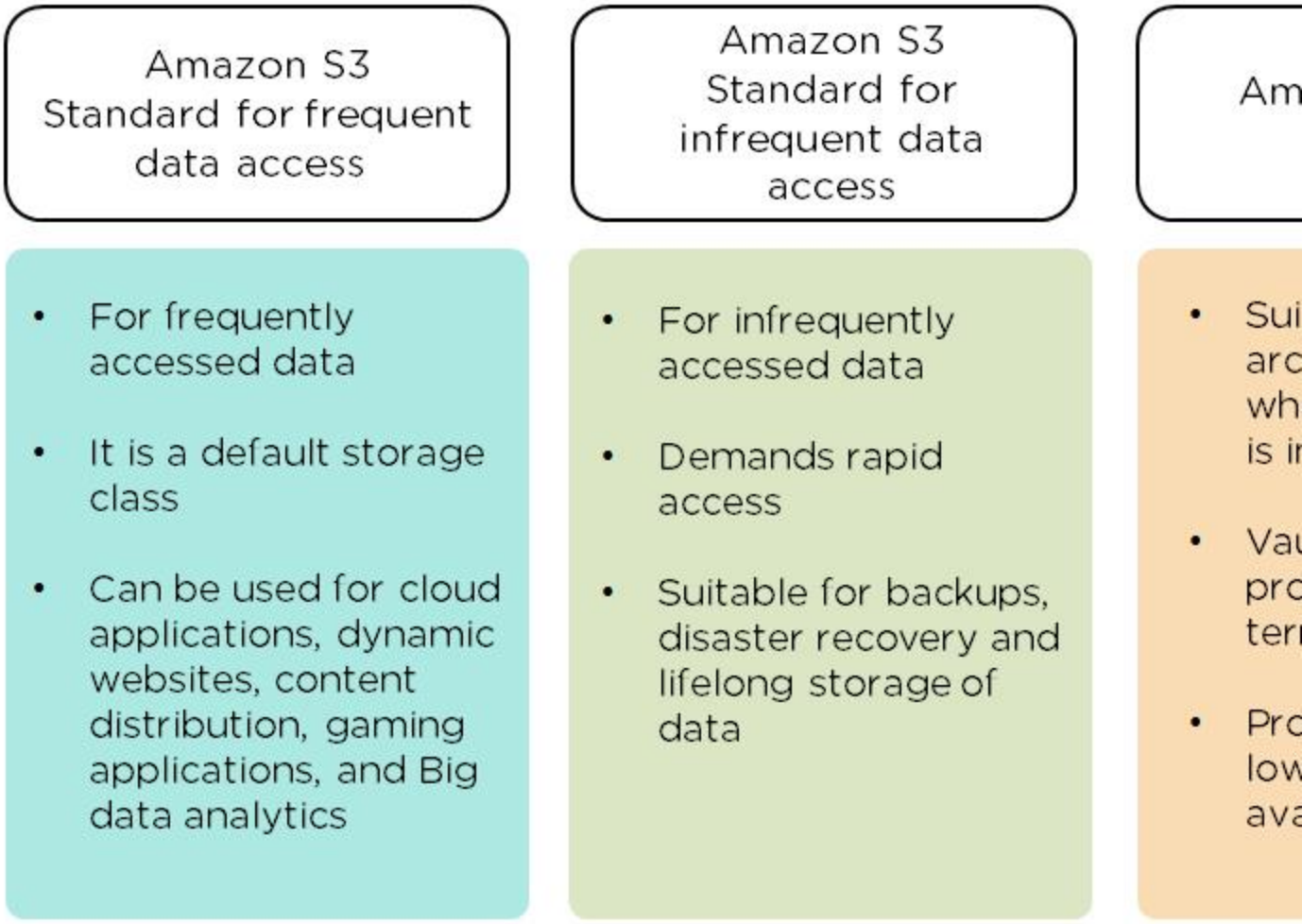
1. Amazon S3 Standard for frequent data access: Suitable for a use case where the latency should below. Example: Frequently accessed data will be the data of students' attendance, which should be retrieved quickly.

2. Amazon S3 Standard for infrequent data access: Can be used where the data is long-lived and less frequently accessed. Example: Students' academic records will not be needed daily, but if they have any requirement, their details should be retrieved quickly.

3. Amazon Glacier: Can be used where the data has to be archived, and high performance is not required. Example: Ex-student's old record (like admission fee) will not be needed daily, and even if it is necessary, low latency is not required.

4. One Zone-IA Storage Class: It can be used where the data is infrequently accessed and stored in a single region. Example: Student's report card is not used daily and stored in a single availability region (i.e., school).

5. Amazon S3 Standard Reduced Redundancy storage: Suitable for a use case where the data is non-critical and reproduced quickly. Example: Books in the library are non-critical data and can be replaced if lost.



A comparison of all storage classes

Storage Class	Durability	Availability	SSL support
STANDARD	99.999999999%	99.99%	Yes
STANDARD_IA	99.999999999%	99.99%	Yes
ONEZONE_IA	99.999999999%	99.5%	Yes
GLACIER	99.999999999%	99.99%	Yes
RRS	99.99%	99.99%	Yes

Technical comparison between classes

Let’s now have a look at the different features offered by S3.

AWS S3 Features

Lifecycle Management

In lifecycle management, Amazon S3 applies a set of rules that define the action to a group of objects. You can manage and store objects in a cost-effective manner. There are two types of actions:

1. Transition Action

With this action, you can choose to move objects to another storage class. With this, you can configure S3 to move your data between various storage classes on a defined schedule. Assume you've got some data stored in the S3 standard class. If this data is not used frequently for 30 days, it would be moved to the S3 infrequent access class. And after 60 days, it is moved to Glacier. This helps you to migrate your data to lower-cost storage as it ages automatically.

2. Expiration Actions

Here, S3 removes all objects within the bucket when a specified date or time period in the object's lifetime is reached.

Bucket Policy

Bucket policy is an IAM policy where you can allow or deny permission to your Amazon S3 resources. With bucket policy, you also define security rules that apply to more than one file within a bucket. For example: If you do not want a user to access the "Simplilearn" bucket, then with the help of JSON script, you can set permissions. As a result, a user would be denied access to the bucket.

Data Protection

Amazon S3 provides IT teams with a highly durable, protected, and scalable infrastructure designed for object storage.

Amazon S3 protects your data using two methods:

1. Data encryption
2. Versioning
3. Cross-region Replication
4. Transfer Acceleration

1. Data Encryption

This refers to the protection of data while it's being transmitted and at rest. It can happen in two ways, client-side encryption (data encryption at rest) and server-side encryption (data encryption in motion).

2. Versioning

It is utilized to preserve, recover, and restore an early version of every object you store in your AWS S3 bucket. Unintentional erases or overwriting of objects can easily be managed with versioning. For example, in a bucket, it is possible to have objects with the same key name but different version IDs.

3. Cross-region Replication

Cross-region replication provides automatic copying of every object uploaded to your buckets (source and destination bucket) in different AWS regions. Versioning needs to be turned on to enable CRR.

4. Transfer Acceleration

This enables fast, easy, and secure transfers of files over long distances between your client and S3 bucket. The edge locations around the world provided by Amazon CloudFront are taken advantage of by transfer acceleration. It works by carrying data over an optimized network bridge that keeps running between the AWS Edge Location (closest region to your clients) and your Amazon S3 bucket.

Aneka frame work

Aneka is Manjrasoft Pty. Ltd.'s solution for developing, deploying, and managing cloud applications. Aneka consists of a scalable cloud middleware that can be deployed on top of heterogeneous computing resources. It offers an extensible collection of services coordinating the execution of applications, helping administrators monitor the status of the cloud, and providing integration with existing cloud technologies. One of Aneka's key advantages is its extensible set of application programming interfaces (APIs) associated with different types of

programming models—such as Task, Thread, and MapReduce—used for developing distributed applications, integrating new capabilities into the cloud, and supporting ...

Overview of Aneka Cloud Application Development Platform shows the basic architecture of Aneka. The system includes four key components, including Aneka Master, Aneka Worker, Aneka Management Console, and Aneka Client Libraries [1]. The Aneka Master and Aneka Worker are both Aneka Containers which represents the basic deployment unit of Aneka based Clouds. Aneka Containers host different kinds of services depending on their role. For instance, in addition to mandatory 5 Figure 2: Basic Architecture of Aneka. The Management Studio and client libraries help in managing the Aneka Cloud and developing applications that utilize resources on Aneka Cloud. The Management Studio is an administrative console that is used to configure Aneka Clouds; install, start or stop Containers; setup user accounts and permissions for accessing Cloud resources; and access monitoring and billing information. The Aneka client libraries, are Application Programming Interfaces (APIs) used to develop applications which can be executed on the Aneka Cloud. Three different kinds of Cloud programming models are available for the Aneka PaaS to cover different application scenarios:: Task Programming, Thread Programming and MapReduce Programming These models represent common abstractions in distributed and parallel computing and provide developers with familiar abstractions to design and implement applications. 2.1.1 Fast and Simple: Task Programming Model Task Programming Model provides developers with the ability of expressing applications as a collection of independent tasks. Each task can perform different operations, or the same operation on different data, and can be executed in any order by the runtime environment. This is a scenario in which many scientific applications fit in and a very popular model for Grid Computing. Also, Task programming allows the parallelization of legacy applications on the Cloud. Thread Programming Model Thread Programming Model offers developers the capability of running multithreaded applications on the Aneka Cloud. The main abstraction of this model is the concept of thread which mimics the semantics of the common local thread but is executed remotely in a distributed environment. This model offers finer control on the execution of the individual components (threads) of an application but requires more management when compared to Task Programming, which is based on a “submit and Client Libraries Programming Models.

1. Task Model

2. Thread Model

3. MapReduce Mode

IBM BLUE CLOUD

History[

IBM cloud computing emerged from the union of its mainframe computing and virtualization technologies. Known as the original virtualization company,[6] IBM's first experiments in virtualization occurred in the 1960s with the development of the virtual machine (VM) on CP-40and CP-67 operating systems. CP-67, a hypervisor used for software testing and development, enabled memory sharing across VMs while giving each user their own virtual memory space. With the machine partitioned into separate VMs, mainframes could run multiple applications and processes at the same time, making the hardware more efficient and cost-effective. IBM began selling VM technology for the mainframe in 1972.

In February 1990, IBM released the RS/6000 (which later became known as POWER processor) based servers. The servers, in combination with the IBM mainframe, were built for complex and mission-critical virtualization. Power systems servers include PowerVMhypervisors with live partition mobility and active memory sharing. Live migration was introduced with POWER6 in May 2007. Next, IBM looked to implement standardization and automation in their technology in order to keep up with the proliferation of data produced by increasingly efficient hardware and data centers. This combination of virtualization, standardization and automation led to the development of IBM cloud computing.

IBM began to develop a strategy for cloud computing in 2007, announcing that it planned to build clouds for enterprise clients and provide services to fill what it regarded as gaps in existing cloud environments. In October 2007, IBM announced a partnership with Google to promote

cloud computing in universities. In addition to donating hardware and machines, the two companies also provided a curriculum to teach students about cloud computing.

IBM claimed in April 2011 that 80% of Fortune 500 companies were using IBM cloud, and that their software and services were used by more than 20 million end-user customers, with clients including American Airlines, Aviva, Carfax, Frito-Lay, IndiaFirst Life Insurance Company, and 7-Eleven.[11]

On 4 June 2013 IBM announced its acquisition of SoftLayer, to form an IBM Cloud Services Division.

By March 4, 2014 IBM acquired Cloudant.

IBM Cloud[edit]

The IBM SmartCloud brand includes infrastructure as a service, software as a service and platform as a service offered through public, private and hybrid cloud delivery models. IBM places these offerings under three umbrellas: SmartCloud Foundation, SmartCloud Services and SmartCloud Solutions.

SmartCloud Foundation consists of the infrastructure, hardware, provisioning, management, integration and security that serve as the underpinnings of a private or hybrid cloud. Built using those foundational components, PaaS, IaaS and backup services make up SmartCloud Services. Running on this cloud platform and infrastructure, SmartCloud Solutions consist of a number of collaboration, analytics and marketing SaaS applications.

IBM also builds cloud environments for clients that are not necessarily on the SmartCloud Platform. For example, features of the SmartCloud platform—such as Tivoli management software or IBM Systems Director virtualization—can be integrated separately as part of a non-IBM cloud platform. The SmartCloud platform consists solely of IBM hardware, software, services and practices.

IBM SmartCloud Enterprise and SmartCloud Enterprise+ compete with products like those of Rackspace and Amazon Web Services. Erich Clementi, vice president of Global Technology Services at IBM, said in 2012 that the goal with SmartCloud Enterprise and SmartCloud

Enterprise+ was to provide an Amazon EC2-like experience primarily for test and development purposes and to provide a more robust experience for production workloads.

In 2011, IBM SmartCloud integrated Hadoop-based InfoSphere BigInsights for big dataGreen Hat for software testing and Nirvanixfor cloud storage. In 2012, the then new CEO Virginia Rometty said the company planned to spend $20 billion on acquisitions by 2015

Users may build their own private cloud or purchase services hosted on the IBM cloud. Users may also purchase IBM hardware, software and services to build their customized cloud environment.

By 2014, the name SmartCloud was replaced with products that have a prefix of IBM Cloud. A product called IBM Cloud Manager with OpenStack was IBM's integration of OpenStack along with a multitude of value additions that would serve enterprise customers. A product called IBM Cloud Orchestrator would serve the orchestration needs of an enterprise. The aforementioned SmartCloud products have been discontinued.

By 2016, the aforementioned product called IBM Cloud Manager with OpenStack was discontinued, although the services organization may be using other versions of OpenStack for large scale cloud deployments.

Public, private and hybrid cloud models

IBM offers cloud delivery options including solely private cloud, solely public cloud, and variations in between. Private, public and hybrid clouds are not strictly distinct, as IBM allows the option to build a customized cloud out of a combination of public cloud and private cloud elements. Companies that prefer to keep all data and processes behind their own firewall can use private cloud services managed by their own IT staff. A company may also choose pay-as-you-go pricing. Hybrid cloud options allow for some processes to be hosted and managed by IBM, while others are kept on a private cloud or on a VPN or VLAN. IBM also offers planning and consultation throughout the deployment process. IBM offers five cloud provision models:

- Private cloud, owned and operated by the customer
- Private cloud, owned by the customer, but operated by IBM (or another provider)
- Private cloud, owned and operated by IBM (or another provider)

- Virtual private cloud services (based on multi-tenanted support for individual enterprises)
- Public cloud services (based on the provision of functions to individuals)[21]

The majority of cloud users choose a hybrid cloud model, with some workloads being served by internal systems, some from commercial cloud providers and some from public cloud service providers.

On August 25, 2011, IBM announced the release of a new hybrid cloud model orchestrated by IBM WebSphere Cast Iron integration of on- and off-premises resources. Enterprises can use Cast Iron integration to link their public cloud appliances— hosted on environments like Amazon EC2, Google Apps, Salesforce.com, Oracle CRM, SugarCRM and a number of others—to their existing systems or in-house, private cloud environments. Cast Iron Integration aims to reduce the time and effort needed for customized coding, in favor of simple workload provisioning through Tivoli Management Framework.

The IBM public cloud offering, SmartCloud Enterprise, was launched on April 7, 2011. SCE is hosted IaaS with service level agreements (SLA)s, and can be offered in a private, public or hybrid model. The environment is hosted on IBM servers (System p or System x), with a standard set of software images to choose from.

For customers who perceive that the security risk of cloud computing adoption is too high, IBM offers private cloud services. IDEAS International wrote in a white paper, "IBM believes that its clients are currently more comfortable with private clouds than public or hybrid clouds, and that many are ready to deploy fundamental business applications in private clouds." For building strictly private clouds, IBM offers IBM Workload Deployer and Cloudburst as ready-to-deploy, “cloud in a box.” Cloudburst provides blade servers, middleware and virtualization for an enterprise to build its own cloud-ready virtual machines. Workload Deployer connects an enterprise's existing servers to virtualization components and middleware in order to help deploy standardized virtual machines designed by IBM

For customers who prefer to perform their own integration of private clouds, IBM offers a choice of hardware and software building blocks, along with recommendations and a reference architecture, prior to deployment. Clients may choose from IBM virtualization-enabled servers, middleware and SaaS applications.

Cloud standards

IBM participates in several cloud standards initiatives within various standards development organizations involved in cloud service models IaaS, PaaS and SaaS, all of which work toward improvements in cloud interoperability and security.

IBM is a member of The Open Group, a council that works for the development of open, vendor-neutral IT standards and certifications. Other members of the group include HP, Oracle, SAP and numerous others.[27] IBM contributed the Cloud Computing Reference Architecture in February 2011 to The Open Group as the basis of an industry-wide cloud architecture. IBM's CCRA is based on real-world input from many cloud implementations across IBM. It is intended to be used as a blueprint/guide for architecting cloud implementations, driven by functional and non-functional requirements of the respective cloud implementation. HP and Microsofthave also published Cloud Computing Reference Architectures.

Within the IaaS space, IBM is a member of the Cloud Management Work Group (CMWG)[within the Distributed Management Task Force (DMTF), which released a draft version of their IaaS APIs, called the Cloud Infrastructure Management Interface (CIMI), on September 14, 2011 The CIMI APIs define a logical model for the management of resources within the Infrastructure as a Service domain. With these APIs, clients can create, manage and connect machines, volumes and networks.

For PaaS and SaaS standards, IBM, Red Hat, Cisco, Citrix, EMC and others contribute to the Topology and Orchestration Specification for Cloud Applications (TOSCA) technical committee within Organization for the Advancement of Structured Information Standards (OASIS), which aims to provide a standardized way of managing the lifecycle of cloud services, for portability of cloud based applications. TOSCA's goal is to advance an interoperability standard that will make it easier to deploy cloud applications without vendor lock-in, while maintaining application requirements for security, governance, and compliance. IBM participates in a number of cloud security related standards including the DMTF Cloud Auditing Data Federation (CADF) working group, and the OASIS Identity in the Cloud (IDCloud) technical committee. CADF is designed to address the need for a cloud provider to provide specific audit event, log and report information on a per-tenant and application basis. IDCloud aims to addresses the serious security

challenges posed by identity management in cloud computing and investigates the need for profiles to achieve interoperability within current standards.

IBM founded the Cloud Standards Customer Council (CSCC) in April 2011, with the Object Management Group (OMG) Kaavo, Rackspace and Software AG, as an end user advocacy group that aimed to accelerate adoption of cloud services and eliminate barriers to security and interoperability associated with the transition to the cloud. In addition to contributing standards requirements to various standards development organizations (SDO), the CSCC also creates guides that companies can use on their own path to cloud adoption.

Distributed search engine

A **distributed search engine** is a search engine where there is no central server. Unlike traditional centralized search engines, work such as crawling, data mining, indexing, and query processing is distributed among several peers in a decentralized manner where there is no single point of control.

History

InfraSearch

In April 2000 several programmers (including Gene Kan, Steve Waterhouse) built a prototype P2P web search engine based on Gnutella called InfraSearch. The technology was later acquired by Sun Microsystems and incorporated into the JXTA projectIt was meant to run inside the participating websites' databases creating a P2P network that could be accessed through the InfraSearch website.

Opencola

On May 31, 2000 Steelbridge Inc. announced development of OpenCOLA a collaborative distributive open source search engine It runs on the user's computer and crawls the web pages and links the user puts in their opencola folder and shares resulting index over its P2P network.

YaCy

On December 15, 2003 Michael Christen announced development of a P2P-based search engine, eventually named YaCy, on the heise online forums.[7][8]

FAROO

In February 2001 Wolf Garbe published an idea of a peer-to-peer search engine started the Faroo prototype in 2004,] and released it in 2005.

Distributed data mining

Featured snippet from the web

Distributed Data Mining (DDM) is a field which deals with analyzing **distributed data** and proposes algorithmic solutions to perform different **data** analysis and **mining** operations in adistributed manner by considering the resource constraints.

Distributed computing and data mining are two elements essential for many commercial and scientific organizations. Data mining is a time and hardware resources consuming process of building analytical models of data. Distribution is often a part of organizations' structure. Authors propose methodology of distributed data mining by combining local analytical models (build in parallel in nodes of a distributed computer system) into a global one without necessity to construct distributed version of data mining algorithm. Different combining strategies are proposed and their verification method as well. Proposed solutions were tested with data sets coming from UCI Machine Learning Repository.