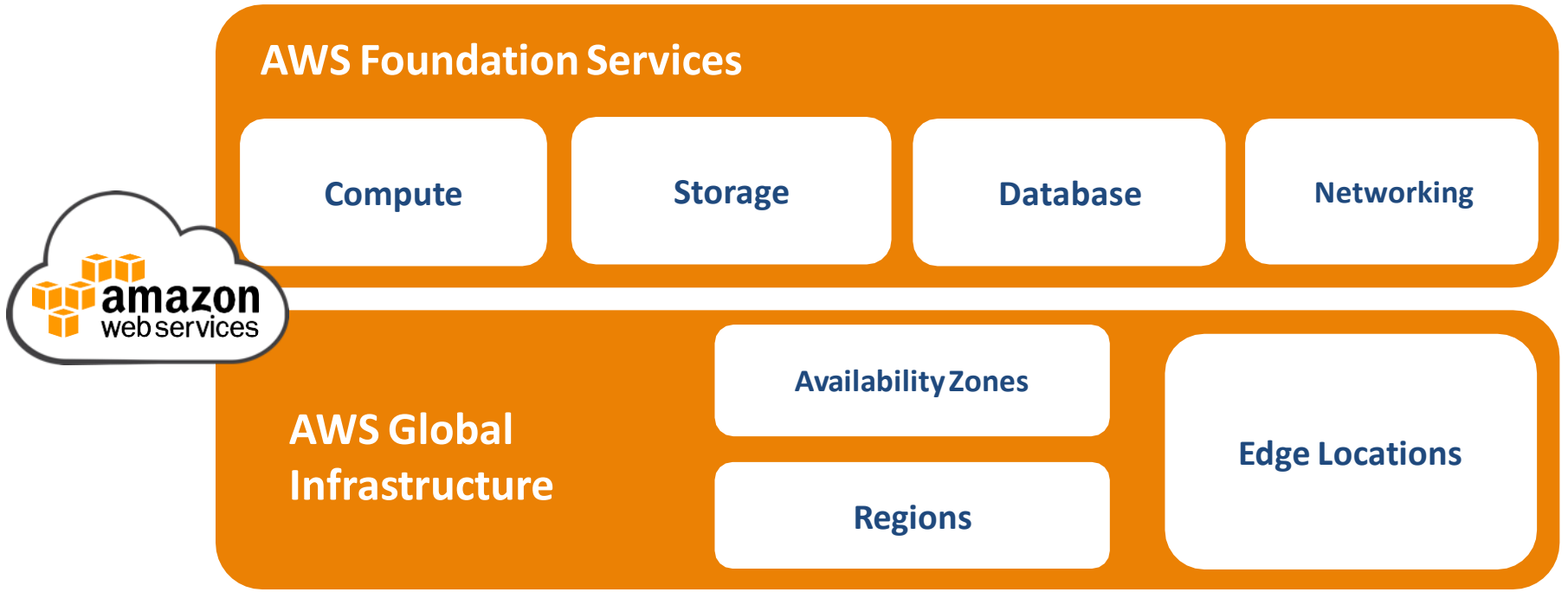
Understanding AWS Storage Options

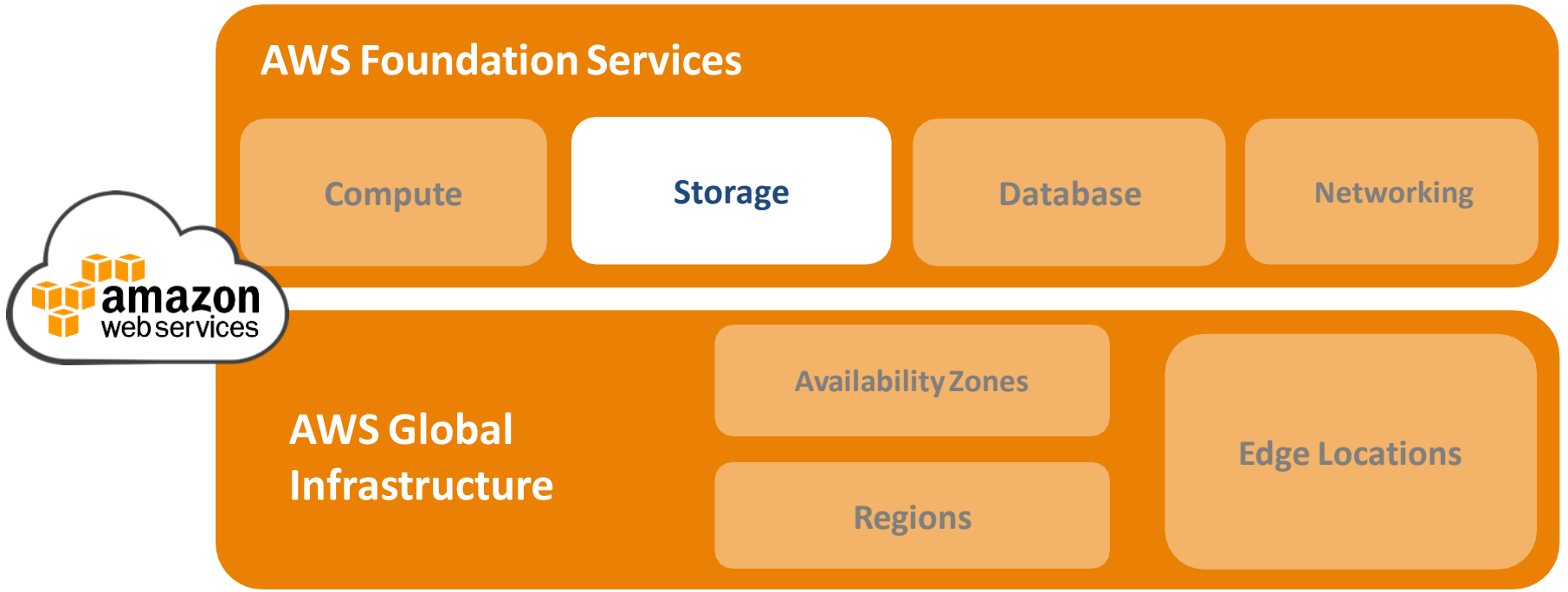
Agenda

- AWS Infrastructure
- What Storage Options should I select?
- How to get my Data inside AWS?
- Demo Time!
- What's Next?

AWS Foundation Services



AWS Foundation Services



AWS Storage Options

- Object storage
- Archive storage
- Block storage
- Gateway solution

Object Storage

Amazon S3

- Amazon S3 is Secure, durable, highly-scalable object storage accessible via a simple web services interface
- It store & retrieve any amount of data for use alone or together with other AWS services

S3

Durable	Available	Low Cost	Scalable
High Performance	Secure	Integrated	Easy to Use
Backup & Archiving	Big Data Analytics	Static Website Hosting	Disaster Recovery
	Content Storage	Distribution Cloud-native Application Data	

A

Amazon S3	Amazon EBS
Web Interface (object storage)	File system interface (block storage)
Scalable	Not easily scalable
Static website hosting	Databases: PostgreSQL, MS SQL, Oracle
Clod Storage / Reduced redundancy storage	Application that require lots of read/write operations

Advantages to Amazon S3

- **Create Buckets** – Create and name a bucket that stores data. Buckets are the fundamental container in Amazon S3 for data storage.

- **Store Data In Buckets** – Store an infinite amount of data in a bucket. Upload as many objects as you like into an Amazon S3 bucket. Each object can contain up to 5 TB of data. Each object is stored and retrieved using a unique developer-assigned key.

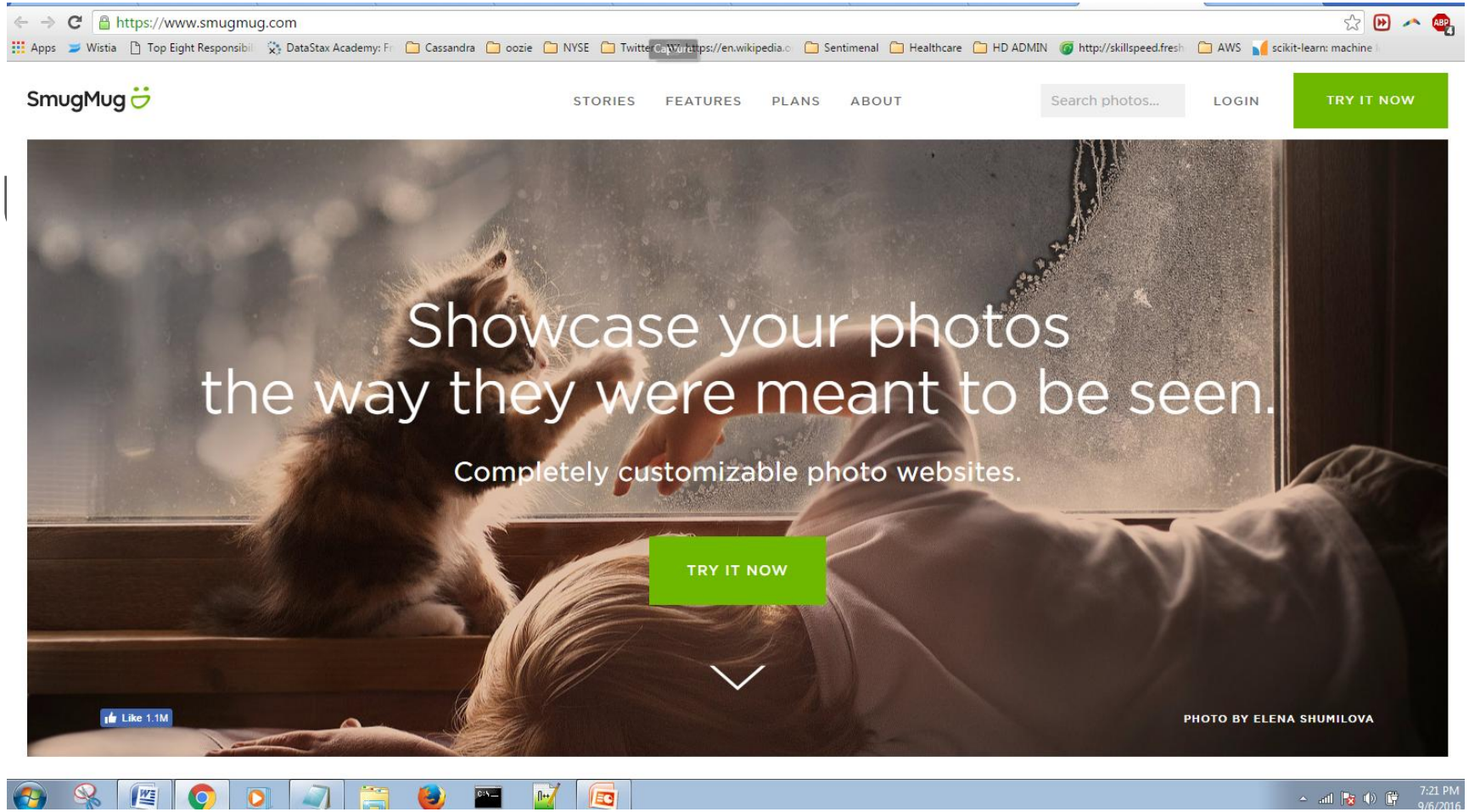
- **Download Data** – Download your data or enable others to do so. Download your data any time you like or allow others to do the same.

- **Permissions** – Grant or deny access to others who want to upload or download data into your Amazon S3 bucket. Grant upload and download permissions to three types of users. Authentication mechanisms can help keep data secure from unauthorized access.

- **Standard Interfaces** – Use standards-based REST and SOAP interfaces designed to work with any Internet-development toolkit.

- SmugMug's Cloud Migration : SmugMug stores billions of photos and images on Amazon S3

S3



- Redfin manages data on hundreds of millions of properties using AWS

- Buckets
- Objects
- Keys
- Regions
- Amazon S3 Data Consistency Model

Amazon S3 Concepts

Buckets & Objects

- A **Bucket** is a **Container** for objects stored in Amazon S3.
- Every Object is contained in a Bucket.
- Objects are the fundamental entities stored in Amazon S3.
- Objects consist of object data and metadata.
- An Object is uniquely identified within a Bucket by a key (name) and a version ID

Purpose

- Organize the Amazon S3 namespace at the highest level,
- Identify the account responsible for storage and data transfer charges,
- Play a role in access control, and serve as the unit of aggregation for usage reporting.

Keys & Regions

▶A **Key** is the unique identifier for an Object within a Bucket.
▶Every Object in a Bucket has **exactly one Key**.

▶Regions - You can choose the geographical region where Amazon S3 will store the Buckets you create.

- **US East (N. Virginia) Region** Uses Amazon S3 servers in Northern Virginia
- **US West (N. California) Region** Uses Amazon S3 servers in Northern California
- **US West (Oregon) Region** Uses Amazon S3 servers in Oregon
- **Asia Pacific (Mumbai) Region** Uses Amazon S3 servers in Mumbai
- **Asia Pacific (Seoul) Region** Uses Amazon S3 servers in Seoul
- **Asia Pacific (Singapore) Region** Uses Amazon S3 servers in Singapore
- **Asia Pacific (Sydney) Region** Uses Amazon S3 servers in Sydney
- **Asia Pacific (Tokyo) Region** Uses Amazon S3 servers in Tokyo
- **EU (Frankfurt) Region** Uses Amazon S3 servers in Frankfurt
- **EU (Ireland) Region** Uses Amazon S3 servers in Ireland
- **South America (São Paulo) Region** Uses Amazon S3 servers in Sao Paulo

Amazon S3 Data Consistency Model

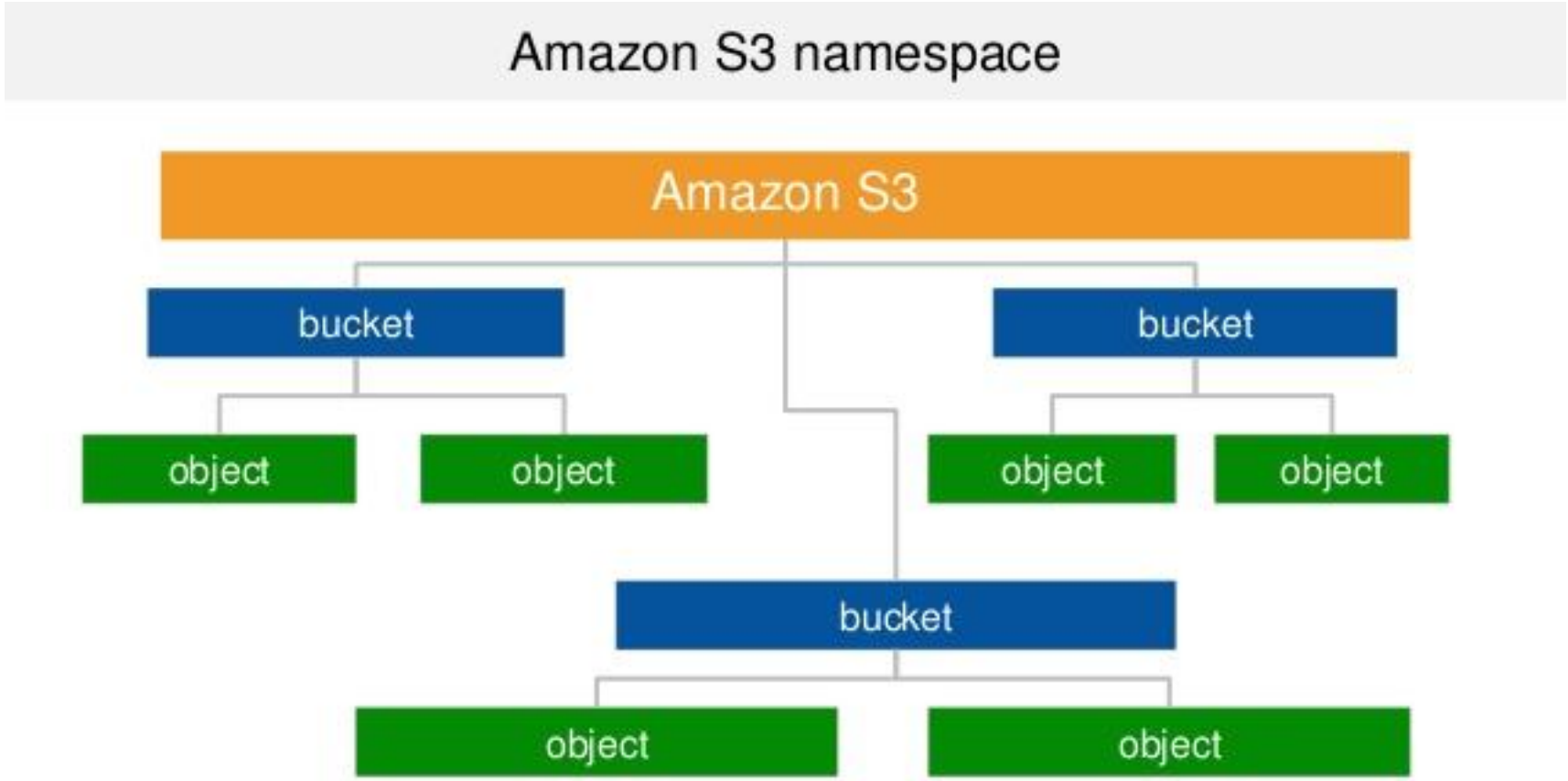
- Amazon S3 provides read-after-write consistency for PUTS of new objects in your S3 bucket in all regions with one caveat.

- Caveat is that if you make a HEAD or GET request to the key name (to find if the object exists) before creating the object, Amazon S3 provides eventual consistency for read-after-write.

- Amazon S3 offers eventual consistency for overwrite PUTS and DELETES in all regions.

- Updates to a single key are atomic.

Am



Protecting Data Using Encryption

- **Use Server-Side Encryption – You request Amazon S3 to encrypt your object before saving it on disks in its data centers and decrypt it when you download the objects.**

- **Use Client-Side Encryption – You can encrypt data client-side and upload the encrypted data to Amazon S3. In this case, you manage the encryption process, the encryption keys, and related tools.**

- **You have the following two options for using data encryption keys:**
  - Use an AWS KMS-managed customer master key
  - Use a client-side master key

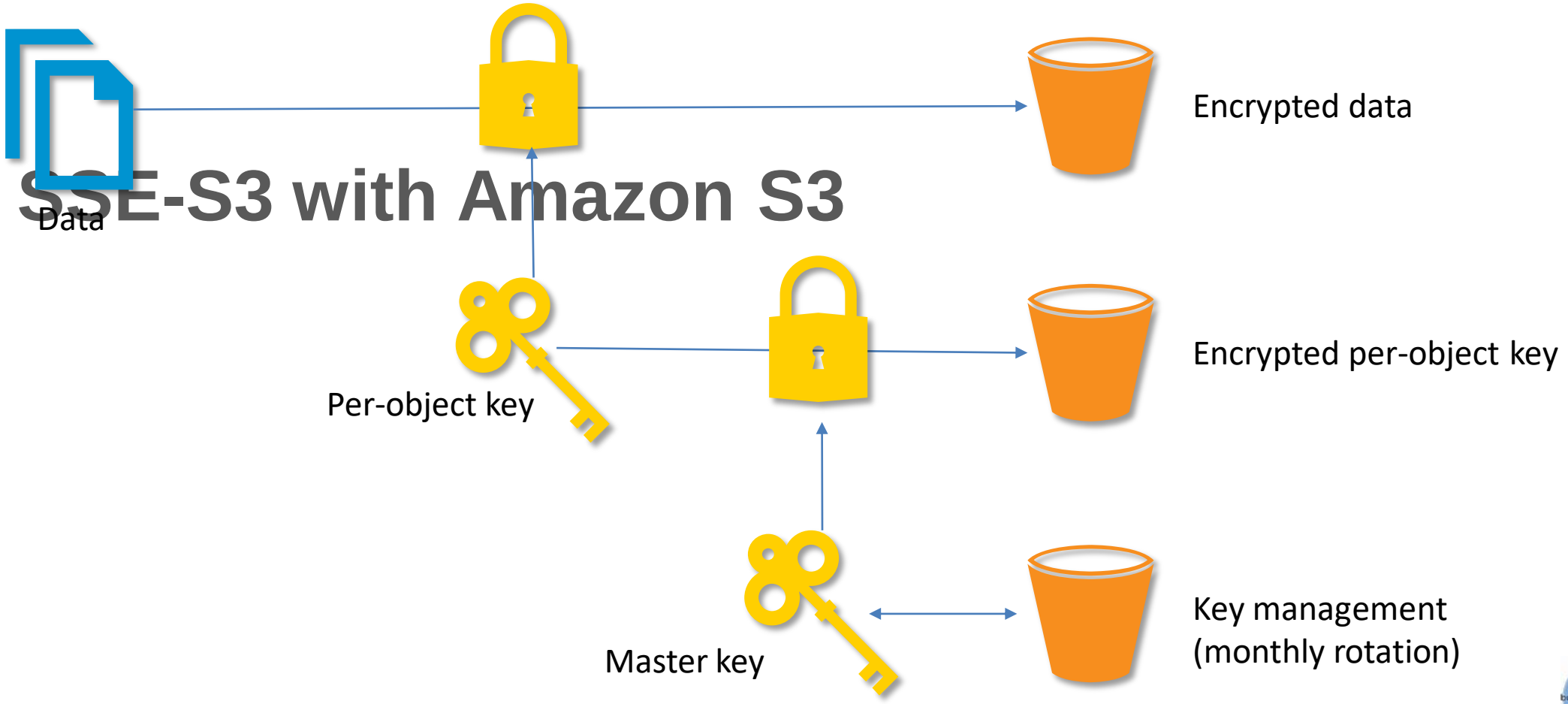
Protecting Data Using Server-Side Encryption

- **Use Server-Side Encryption with Amazon S3-Managed Keys (SSE-S3)** - Each object is encrypted with a unique key employing strong multi-factor encryption.

- **Use Server-Side Encryption with AWS KMS-Managed Keys (SSE-KMS)** - Similar to SSE-S3, but with some additional benefits along with some additional charges for using this service. There are separate permissions for the use of an envelope key (that is, a key that protects your data's encryption key) that provides added protection against unauthorized access of your objects in S3.

- **Use Server-Side Encryption with Customer-Provided Keys (SSE-C) –** You manage the encryption keys and Amazon S3 manages the encryption, as it writes to disks, and decryption, when you access your objects.

With SSE-S3, Amazon S3 will encrypt your data at rest and manage the encryption keys for you



- Reduced redundancy storage is designed to provide 99.99% durability of objects over a given year.

- Reduced redundancy storage stores objects on multiple devices across multiple facilities, providing 400 times the durability of a typical disk drive, but it does not replicate objects as many times as Amazon S3 standard storage

Reduced Redundancy Storage

▶To set the storage class of an object you upload to RRS, you set **x-amz-storage** class to **REDUCED_REDUNDANCY** in a **PUT** request.

▶The following example sets the storage class of my-image.jpg to RRS.

S
U

```
PUT /my-image.jpg HTTP/1.1
Host: myBucket.s3.amazonaws.com
Date: Wed, 12 Oct 2009 17:50:00 GMT
Authorization: AWS AKIAIOSFODNN7EXAMPLE:xQE0diMbLRepdf3YB+FIEXAMPLE=
Content-Type: image/jpeg
Content-Length: 11434
Expect: 100-continue
x-amz-storage-class: REDUCED_REDUNDANCY
```

- We can also change the storage class of an object that is already stored in Amazon S3 by copying it to the same key name in the same bucket.
- To do this, we use the following request headers in a PUT Object copy request:
  - *x-amz-metadata-directive* set to COPY
  - *x-amz-storage-class* set to STANDARD, STANDARD_IA, or REDUCED_REDUNDANCY

Changing the Storage Class of an Object in Amazon S3

In this lesson we understood

- Amazon S3

Wrap – up

scmGalaxy

SCM-DevOps-Build-Release

amazon web services

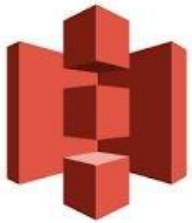
Simple Storage Service (S3)

Features:

- S3 is Simple Storage Service
- Amazon S3 provides unlimited storage space and works on the pay as you use model. Service rates gets cheaper as the usage volume increases
- Amazon S3 is an Object level storage (not a Block level storage) and cannot be used to host OS or dynamic websites
- Amazon S3 resources (for example buckets and objects) are private by default
-

amazon web services

Simple Storage Service (S3)



Buckets

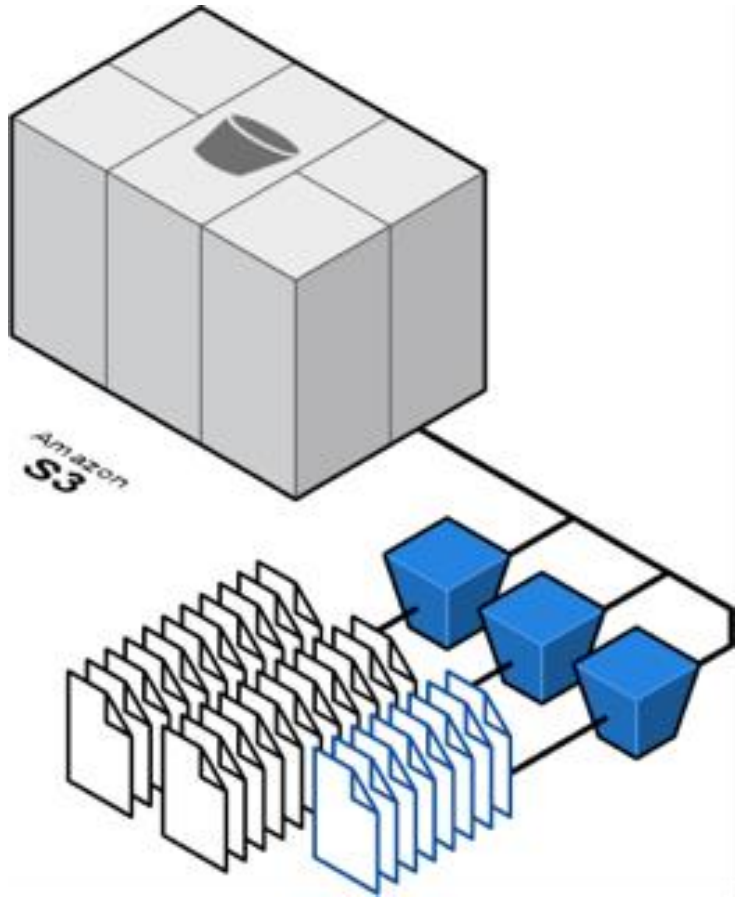
- A bucket is a container for objects stored in Amazon S3 and help organize the Amazon S3 namespace.
- A bucket is owned by the AWS account that create it and helps identify the account responsible for storage and data transfer charges
- Amazon S3 bucket names are globally unique, regardless of the AWS region in which you create the bucket
- Even though S3 is a global service, Amazon S3 buckets are created within a region specified during the creation of the bucket
- Every object is contained in a bucket and there is no limit on the number of objects that a bucket can have

Simple Storage Service (S3

Objects:

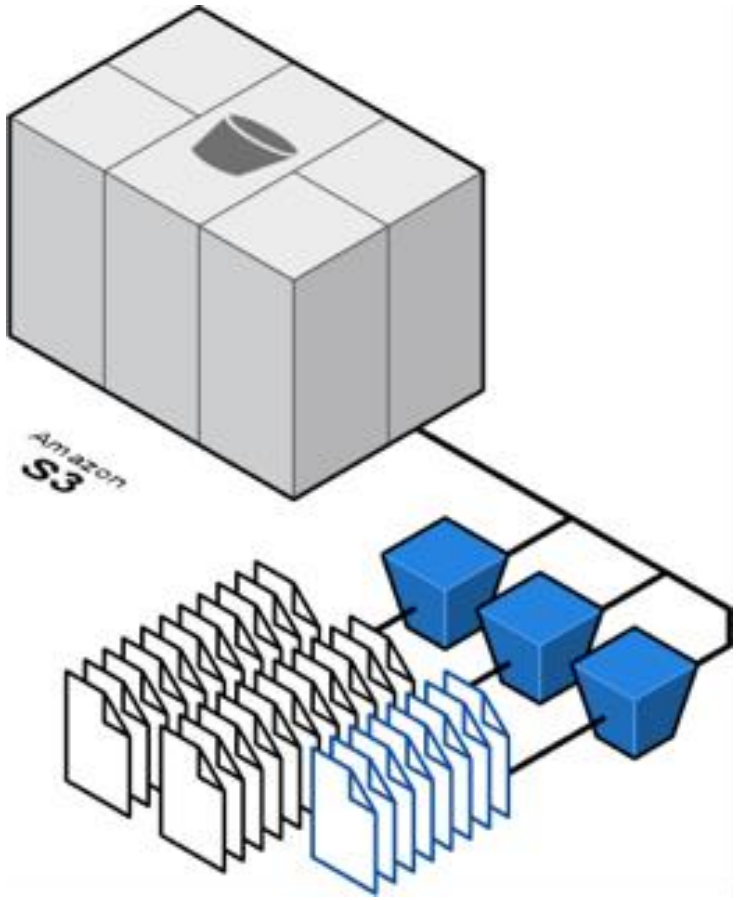
- Objects are the fundamental entities stored in Amazon S3.
- Object is uniquely identified within a bucket by a key (name) and a version ID.
- Objects consist of object data, metadata and others
    - **Value** is Data portion is opaque to Amazon S3.
    - **Metadata** is the data about the data and is a set of name-value pairs that describe the object for e.g. content-type, size, last modified. You can also specify custom metadata at the time the object is stored.
    - **Key** is object name
    - **Version ID** is the version id for the object and in combination with the key helps to unique identify an object within a bucket
    - **Subresources** helps provide additional information for an object
    - **Access Control Information** helps control access to the objects stored in S3

Simple Storage Service



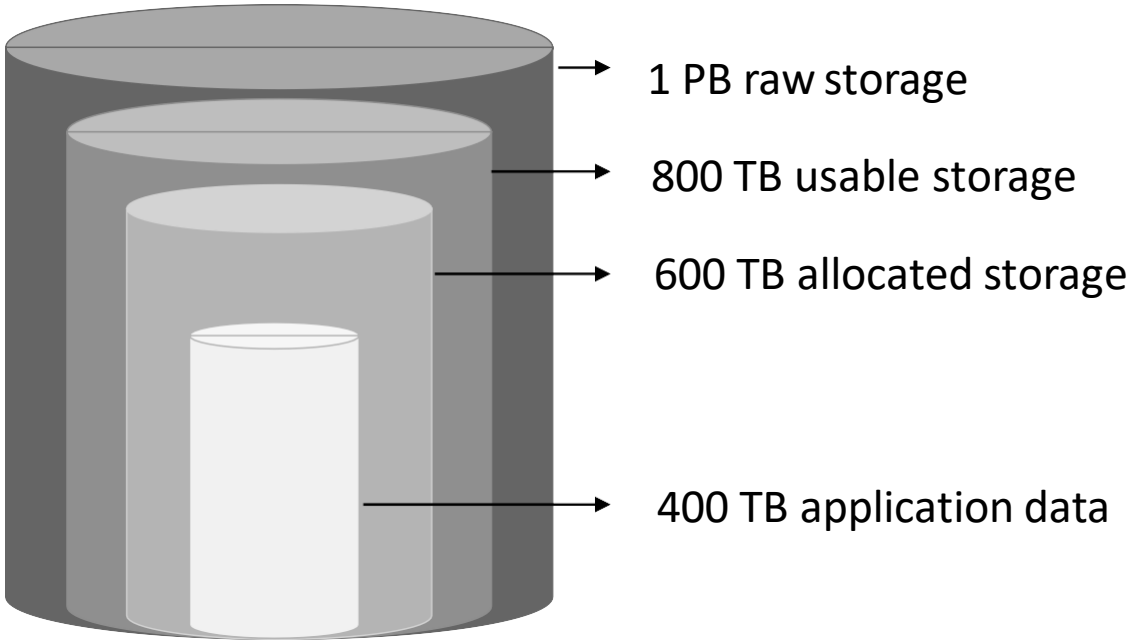
- Internet accessible storage via HTTP / HTTPS
- Highly scalable, fully managed object storage
- Virtually unlimited storage capacity
- 1 byte to 5 TB in size per object
- Trillions of unique customer objects
- Millions of transactions per second
- 99.999999999% durability
- First AWS service introduced in 2006
- Audio, video, Images Backups etc.
- Unlimited Bucket Size
- Priced on storage used and transfer out
- Its not a file system
- 99.99% SLA

Simple Storage Service



- Can use HTTP/S endpoints to store and retrieve any amount of data, at any time, from anywhere on the web
- Auditing is provided by access logs
- Provides standards-based REST and SOAP interfaces
- Can use optional server-side encryption using AWS or customer-managed provided client-side encryption

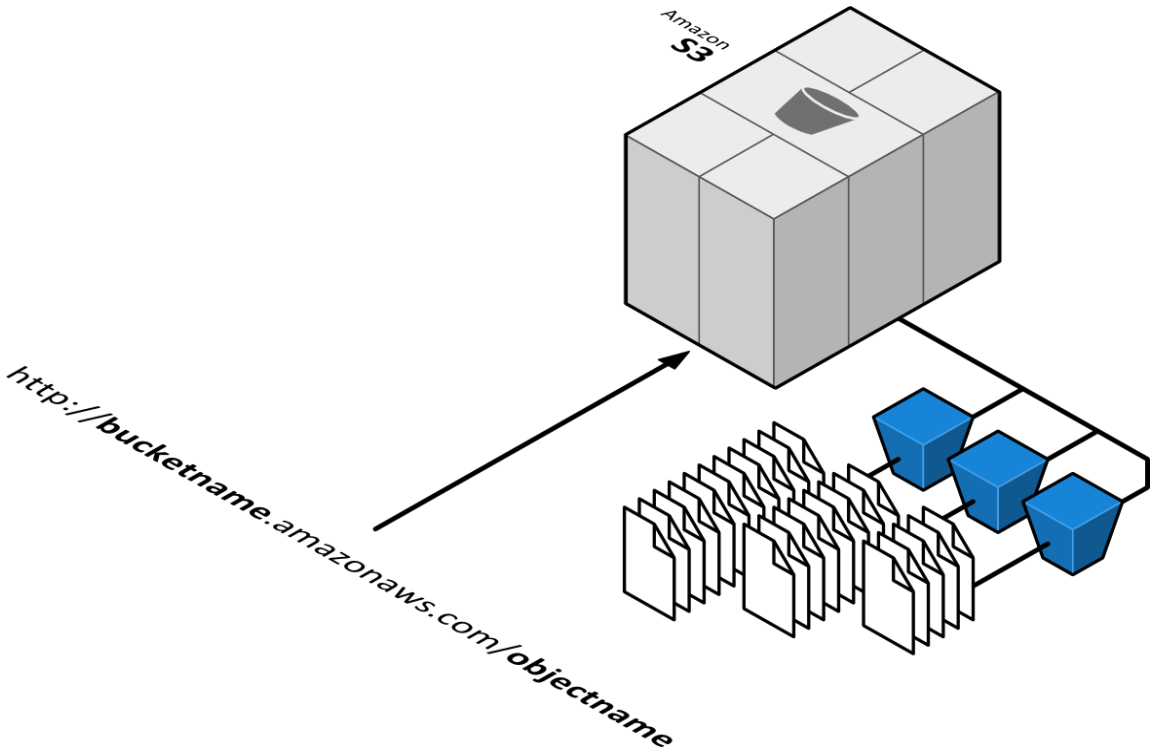
Simple Storage Service



Traditional Storage

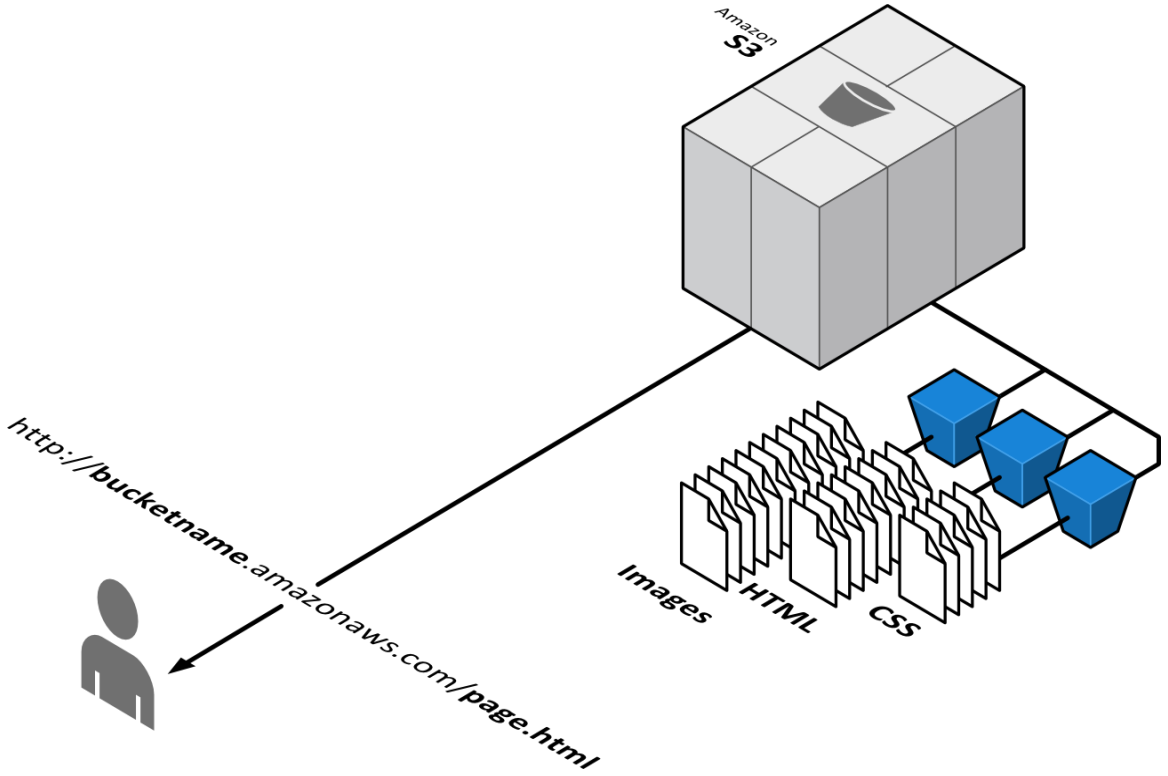
Amazon S3

S3 – Buckets and Objects



- Fully managed
- Store in buckets
- Versioning
- Access control lists and bucket policies
- AES-256 bit encryption at rest
- Private by default
- Addressable over the internet if public
- Ideal for images, videos, application data, backups and more

S3 – Static Content Website



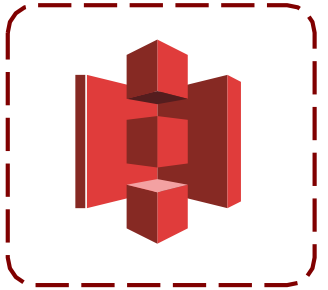
- S3 becomes your webserver
- Offload static content to S3 and run dynamic content on EC2

S3 – When to use?

Use Amazon S3 when you need

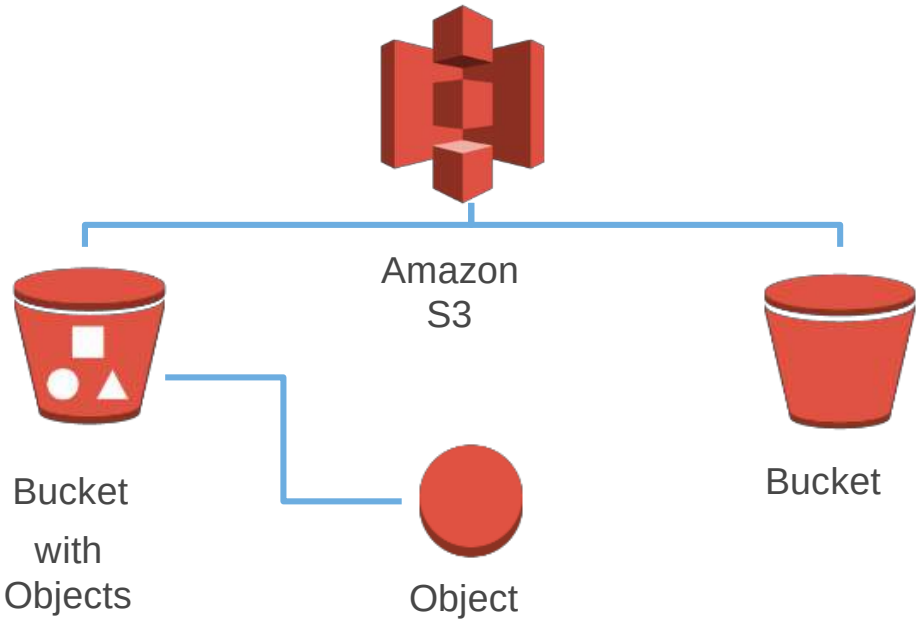
- Web-scale storage capacity and performance for web applications
- High data durability
- Storage for log files
- Storage for backup and active archives
- Single-origin store with delivery through Content Delivery Networks such as Amazon CloudFront
- Ingestion point for Big Data application

Common Use Scenarios



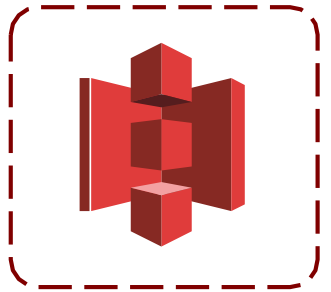
- Storage and backup
- Application file hosting
- Media hosting
- Software delivery
- Store AMIs and snapshots

Amazon S3 Concepts



- Amazon S3 stores data as objects within **buckets**
- An object is composed of a file and optionally any **metadata** that describes that file
- You can have **up to 100 buckets** in each account
- You can **control access** to the bucket and its objects

Object Keys



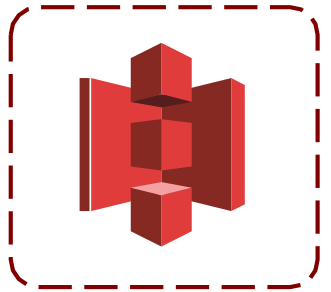
An object key is the unique identifier for an object in a bucket.

http://doc.s3.amazonaws.com/2006-03-01/AmazonS3.html

Bucket

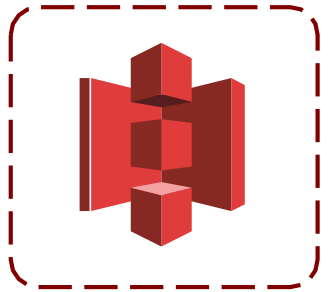
Object/Key

Amazon S3 Security



- You can **control access** to buckets and objects with:
  - Access Control Lists (ACLs)
  - Bucket policies
  - Identity and Access Management (IAM) policies
- You can upload or download data to Amazon S3 via **SSL** encrypted endpoints.
- You can **encrypt data** using AWS SDKs.

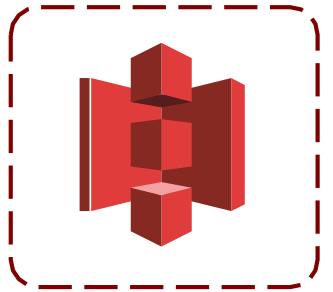
Amazon S3 Versioning



- Protects from **accidental overwrites and deletes** with no performance penalty.
- Generates a **new version with every upload**.
- Allows easily retrieval of deleted objects or **roll back** to previous versions.
- Three states of an Amazon S3 bucket

  - Un-versioned (default)
  - Versioning-enabled
  - Versioning-suspended

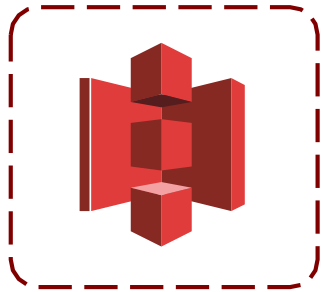
Amazon S3 Object Lifecycle



Lifecycle management defines how Amazon S3 manages objects during their lifetime. Some objects that you store in an Amazon S3 bucket might have a well-defined lifecycle:

- Log files
- Archive documents
- Digital media archives
- Financial and healthcare records
- Raw genomics sequence data
- Long-term database backups
- Data that must be retained for regulatory compliance

Amazon S3 Pricing



- Pay only for what you use
- No minimum fee
- Prices based on location of your Amazon S3 bucket
- Estimate monthly bill using the **AWS Simple Monthly Calculator**
- Pricing is available as:
  - Storage Pricing
  - Request Pricing
  - Data Transfer Pricing: data transferred out of Amazon S3

Amazon S3 Storage Classes

Storage Class	Durability	Availability	Other Considerations
Amazon S3 Standard	99.999999999%	99.99%	
Amazon S3 Standard - Infrequent Access (IA)	99.999999999%	99.9%	• Retrieval fee associated with objects • Most suitable for infrequently accessed data
Glacier	99.999999999%	99.99% (once restored)	• Not available for real-time access • Must restore objects before you can access them • Restoring objects can take 3-5 hours

Two Types of S3 Storage

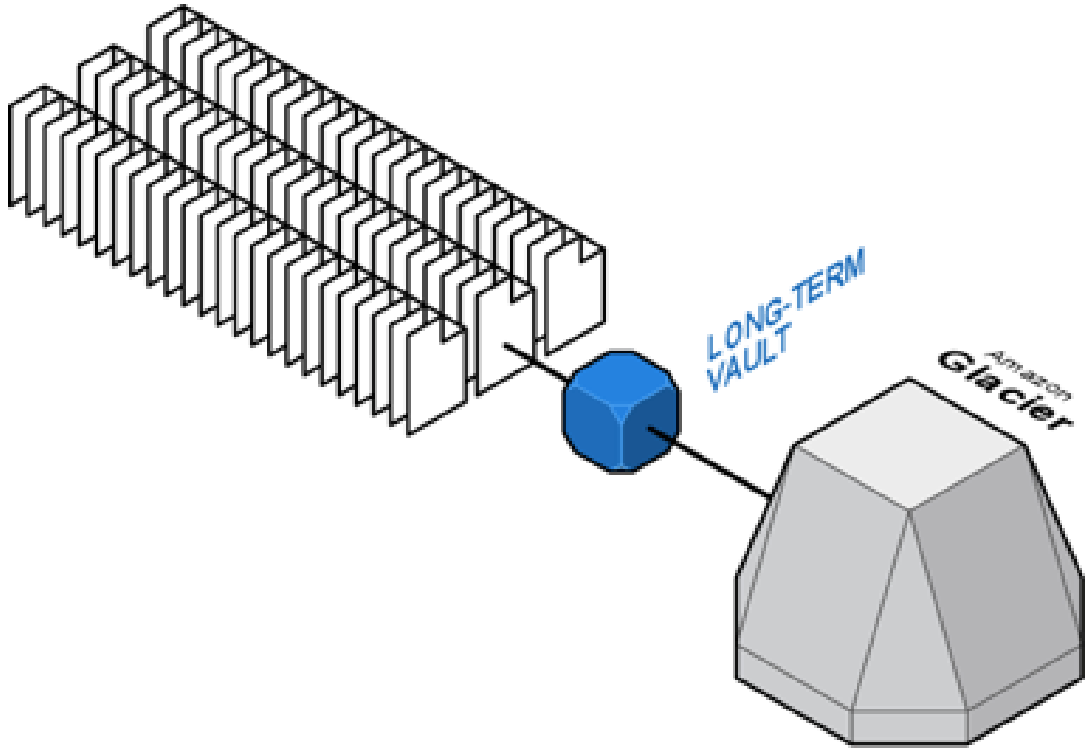
- Standard storage
  - 99.999999999% durability
  - First 1TB $0.0300 / GB
- Reduced Redundancy Storage (RRS)
  - First 1TB $0.0240 / GB
  - Reduced durability - 99.99%
- Granular storage type selection

- Uses Amazon S3 to store daily logs and images
- Generates more than 1.2 TBs of logs per day
- Runs ~250 Hadoop jobs per day, processing more than 30TBs of data

amazon web services | intel

Archival Storage

Amazon Glacier



- Long term low-cost archiving service
- Optimal for infrequently accessed data
- 99.999999999% durability
- 3-5 hours retrieval time
- $0.01 per GB / month
- $120 per TB / year

Storage Cost

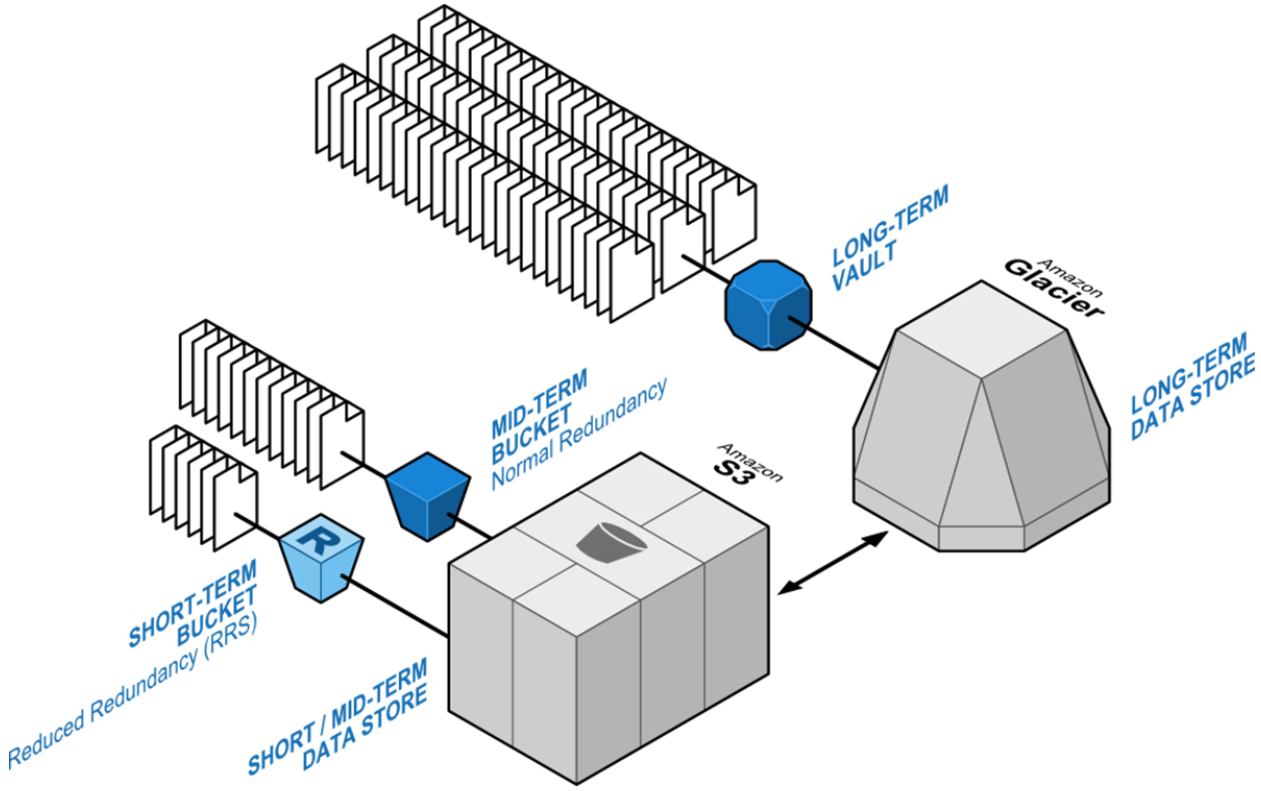
vs.

Retrieval Cost

Amazon S3 and Amazon Glacier Integration

Policy-based archiving service

Amazon S3 Lifecycle Policies



- S3 Lifecycle policies allow to delete or move objects based on age
- Set rules per S3 bucket

- Example:
  - Move object to Glacier after 30 days
  - Delete object after 365 days

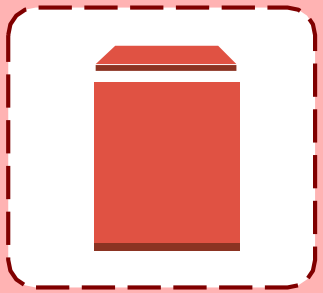
Use Case: SoundCloud

- One of the world's leading social sound platform
- Audio files must be transcoded and stored in multiple formats
- Stores 2.5 PBs of data
- Transcoded files served from Amazon S3
- Originals moved to Amazon Glacier for cost savings

Block Storage


code
deploy
plan
release
build
operate
test
monitor

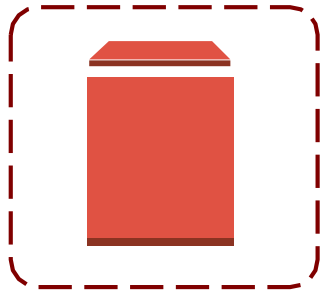
Amazon Elastic Block Store (EBS)



Amazon
EBS

- **Persistent block level storage** volumes offer consistent and low-latency performance.
- Stored data is automatically replicated within its Availability Zone.
- Snapshots are stored durably in Amazon S3.

Amazon EBS Lifecycle



Call CreateVolume 1 GiB to 16 TiB

Vast amounts of unused space

Create

Attach

Call AttachVolume to affiliate with one Amazon EC2 instance

Attached and In Use

- Format from Amazon EC2 instance OS
- Mount formatted drive

CreateSnapshot

Snapshot to Amazon S3

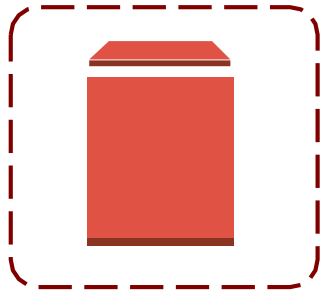
Detach

Call DetachVolume

Deleted

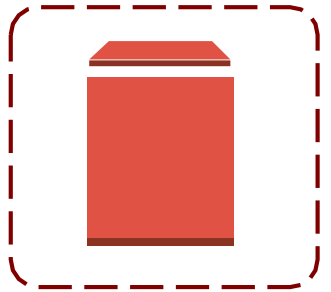
Call DeleteVolume

Amazon EBS Volume Types



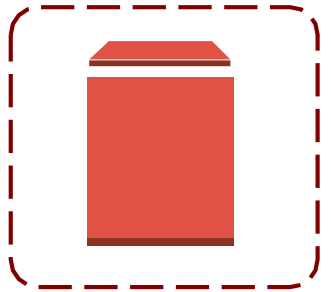
- SSD-backed volumes are
  - Optimized for **transactional** workloads that involve **frequent read/write** operations with **small I/O** size.
  - Dominant in **IOPS** performance.
- HDD-backed volumes are
  - Optimized for **large streaming** workloads.
  - Dominant in **throughput** (measured in MiB/s).

Amazon EBS Volume Types



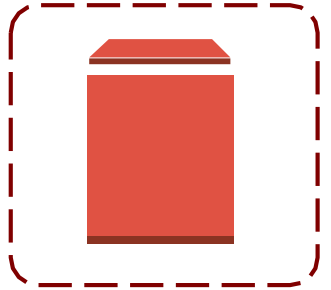
	SSD		HDD	
Volume Type	**General Purpose SSD (gp2)**	**Provisioned IOPS SSD (io1)**	**Throughput Optimized HDD (st1)**	**Cold HDD (sc1)**
Description	Balances price and performance for a wide variety of **transactional** loads.	**Highest-performance** SSD volume designed for **mission-critical** applications.	**Low-cost** HDD designed for **frequently accessed**, **throughput-intensive** workloads.	**Lowest cost** HDD designed for less frequently accessed workloads.
Volume Sizes	1 GiB – 16 TiB	4 GiB – 16 TiB	500 GiB – 16 TiB	500 GiB – 16 TiB
Dominant Performance Attribute	IOPS	IOPS	MiB/s	MiB/s

Amazon EBS Facts



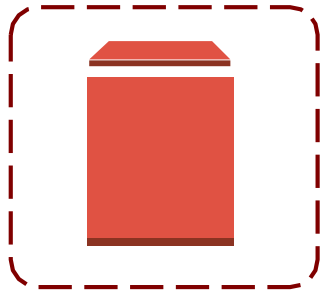
- EBS is recommended when data must be **quickly accessible** and requires **long-term persistence**.
- You can launch your EBS volumes as **encrypted** volumes – data stored at rest on the volume, disk I/O, and snapshots created from the volume are all encrypted.
- You can create **point-in-time snapshots** of EBS volumes, which are persisted to Amazon S3.

Amazon EBS Use Cases



- **OS:** Use for boot/root volume, secondary volumes
- **Databases:** Scales with your performance needs
- **Enterprise applications:** Provides reliable block storage to run mission-critical applications
- **Business continuity:** Minimize data loss and recovery time by regularly backing up using EBS Snapshots
- **Applications:** Install and persist any application

Amazon EBS Pricing

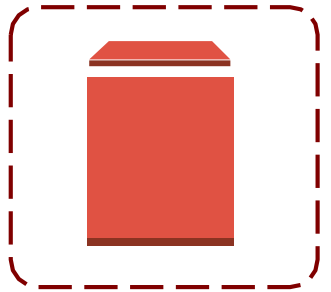


Pay for what you provision:

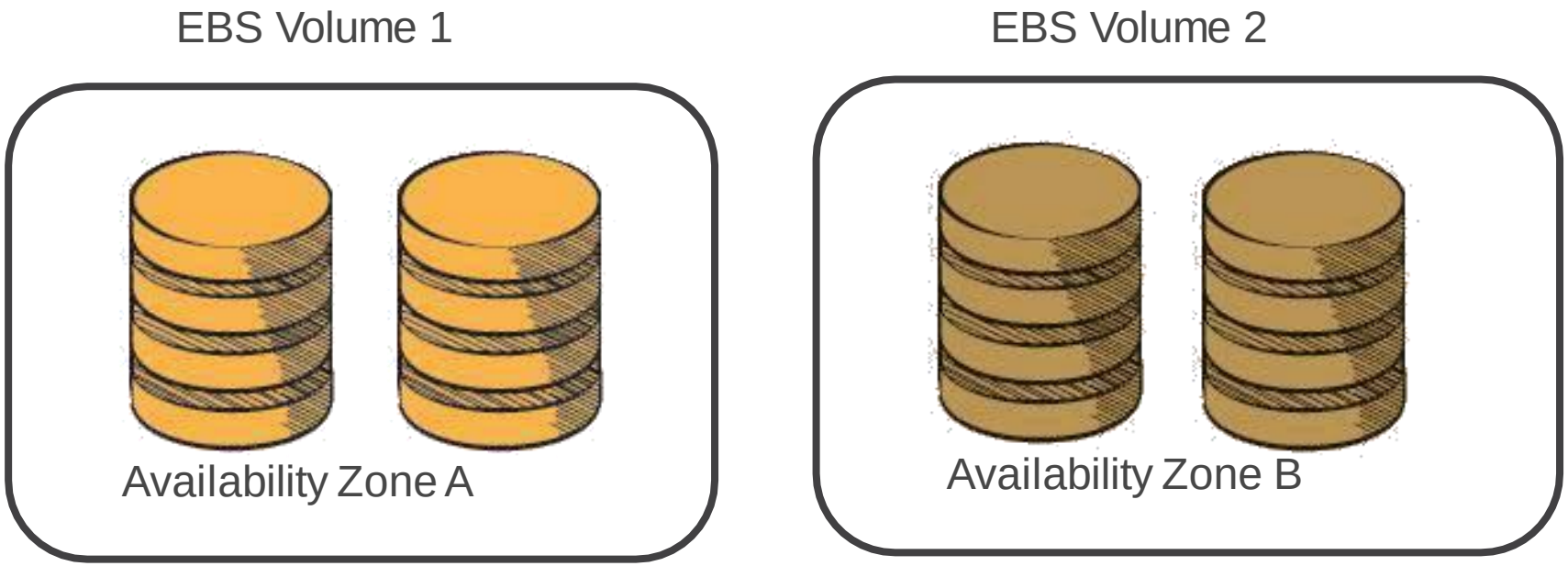
- Pricing based on region
- Review Pricing Calculator online
- Pricing is available as:
  - Storage
  - IOPS

** Check Amazon EBS Pricing page for current pricing for all regions.*

Amazon EBS Scope

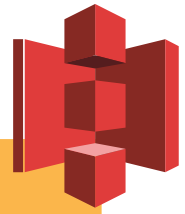


Amazon EBS volumes are in a single Availability Zone



Volume data is replicated across multiple servers in an Availability Zone.

Amazon EBS and Amazon S3



	Amazon EBS	Amazon S3
Paradigm	Block storage with file system	Object store
Performance	Very fast	Fast
Redundancy	Across multiple servers in an Availability Zone	Across multiple facilities in a Region
Security	EBS Encryption – Data volumes and Snapshots	Encryption
Access from the Internet?	No (1)	Yes (2)
Typical use case	It is a disk drive	Online storage

(1) Accessible from the Internet if mounted to server and set up as FTP, etc.
(2) Only with proper credentials, unless ACLs are world-readable

Amazon EC2 Instance Storage

- Is local, complimentary **direct attached block storage**.
- Includes availability, number of disks, and size **based on EC2 instance type**.
- Is optimized for **up to 365,000 Read IOPS** and 315,000 First Write IOPS.
- Is SSD or magnetic.
- Has **no persistence**.
- A**utomatically deletes** data when an EC2 instance stops, fails or is terminated.

Amazon EBS vs. Amazon EC2 Instance Store

Amazon EBS

- Data stored on an Amazon EBS volume can persist independently of the life of the instance.
- Storage is **persistent**.

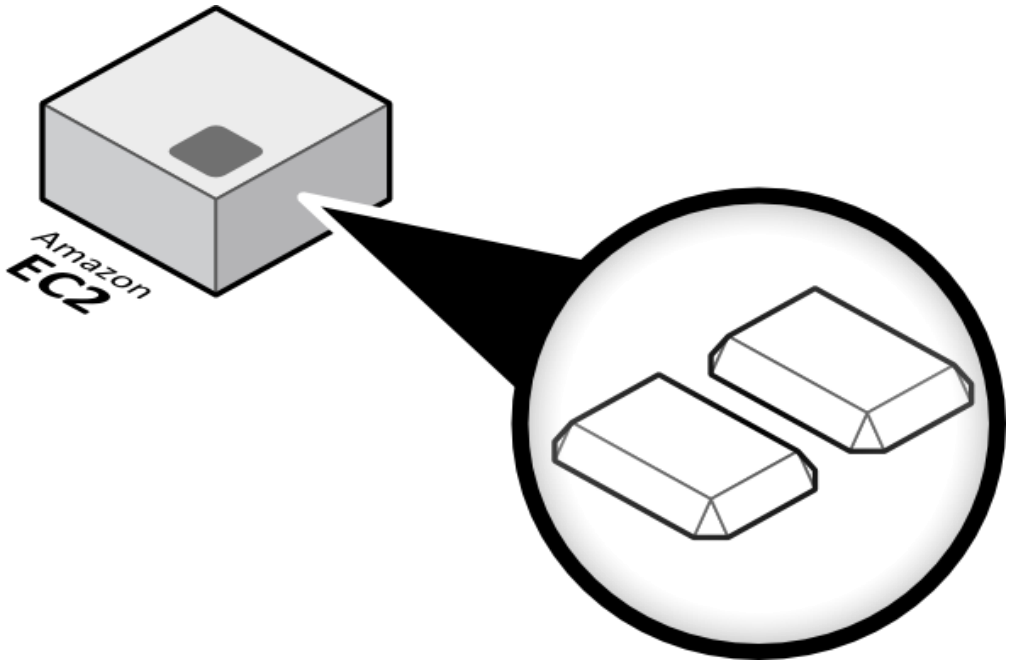
Amazon EC2 Instance Store

- Data stored on a local instance store persists only as long as the instance is alive.
- Storage is **ephemeral**.

Reboot vs. Stop vs. Terminate

Characteristic	Reboot	Stop/Start (EBS-backed instances only)	Terminate
Host computer	The instance **stays on the same host computer**.	The instance runs on a **new host computer**.	
Public IP address	No change	**New address assigned**	
Elastic IP addresses (EIP)	EIP remains associated with the instance.	EIP remains associated with the instance.	EIP is **disassociated** from the instance.
Instance store volumes	Preserved	**Erased**	**Erased**
EBS volume	Preserved	Preserved	Boot volume is **deleted by default**.
Billing	Instance billing hour doesn't change.	You **stop incurring charges** as soon as state is changed to *stopping*.	You **stop incurring charges** as soon as state is changed to *shutting-down*.

EC2 Instance Storage

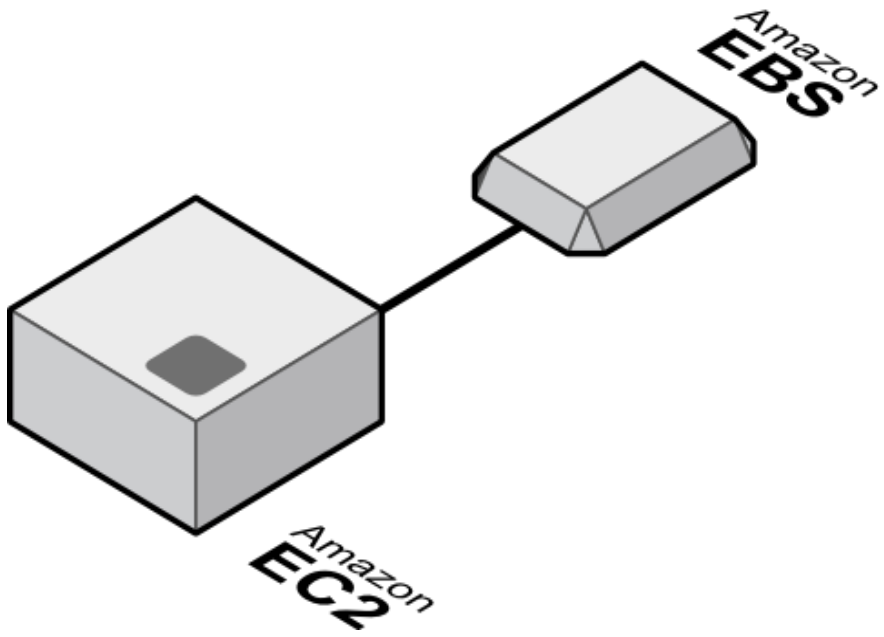


- Every EC2 comes with instance storage
- Temporary block-level storage
- Free storage with EC2 instance
- There is no SLA, access speed is not guaranteed
- Local, direct attached resource
- Size is based on EC2 instance type
- Storage optimized instances for up to 365,000 r/s and 315,000 w/s
- Replicated data for load-balanced web servers
- Choice of SSD or magnetic

It's volatile!

- No persistence
- All data is automatically deleted when an EC2 instance stops, fails or is terminated

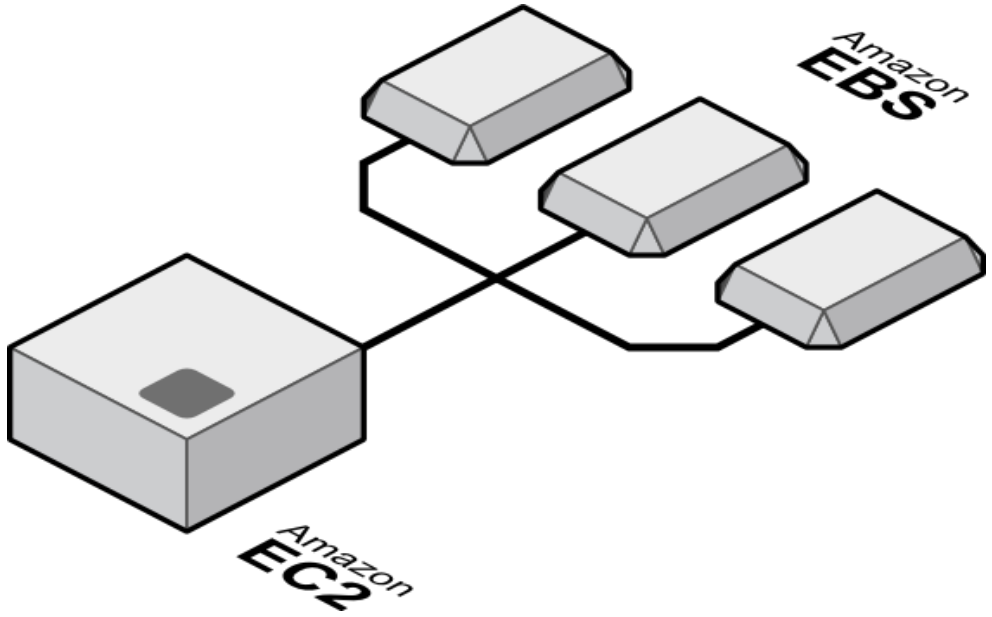
Elastic Block Store



- High performance block storage
- Persistent block storage
- 1GB to 6TB in size *
- Mount as drives to EC2 instances
- Does not need to be attached to an instance
- SSD or Magnetic
- Burstable or provisioned throughput
- EBS leverages S3 for snapshot storage
- Supports incremental snapshots
- Can be transferred between AZ
- Not internet accessible

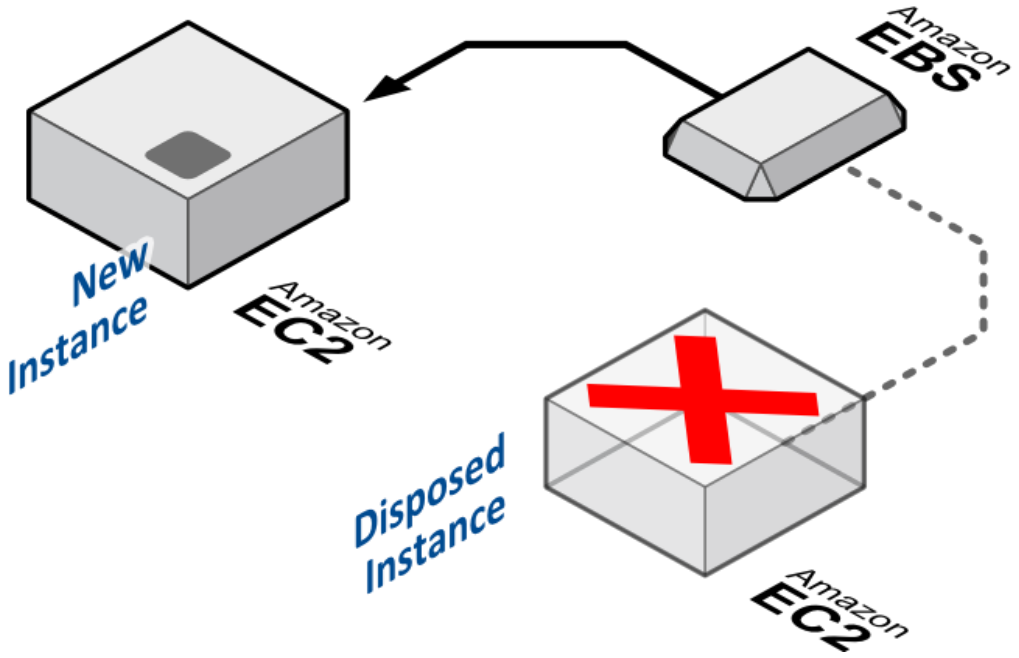
Upcoming: Larger and faster EBS volumes for up to 16TB

Elastic Block Store



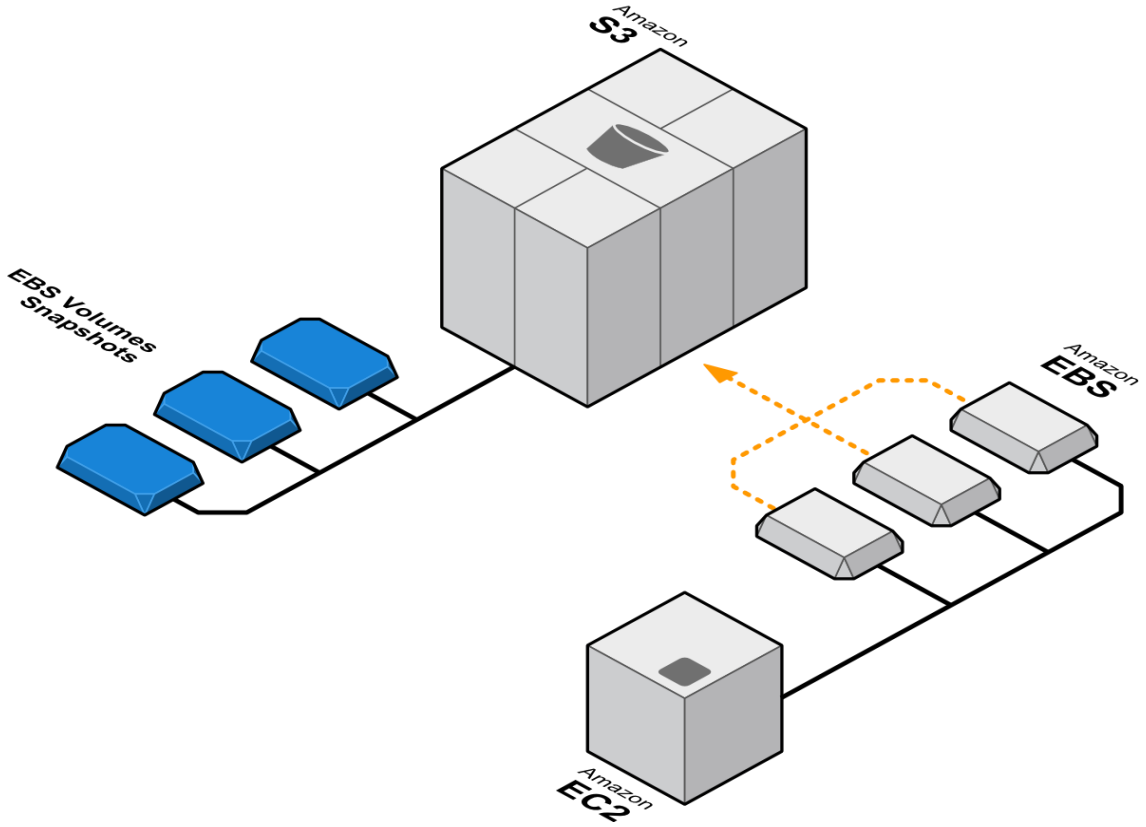
- You can attach multiple EBS volumes
- RAID to increase performance or storage capacity

Elastic Block Store



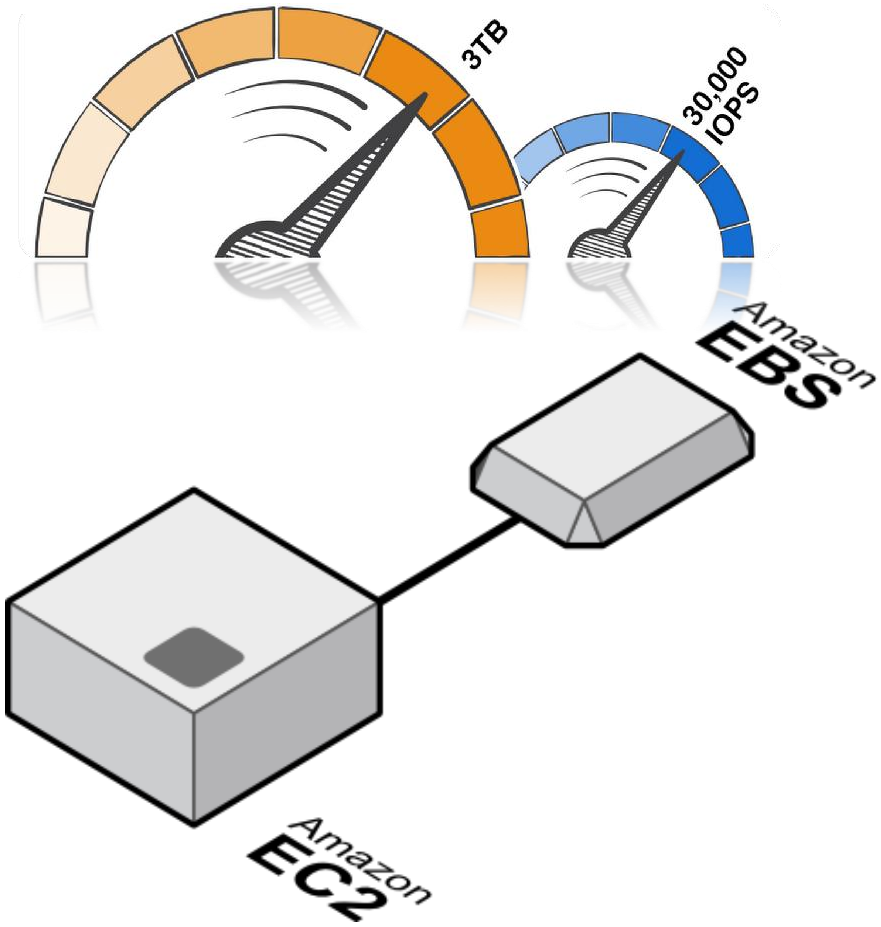
- You can reattach your EBS volume to a new instance
- Data is persisted

Elastic Block Store



- You can snapshot your EBS volume into our highly durable storage service
- Create new EBS volumes from snapshots or clone drives

EBS Performance

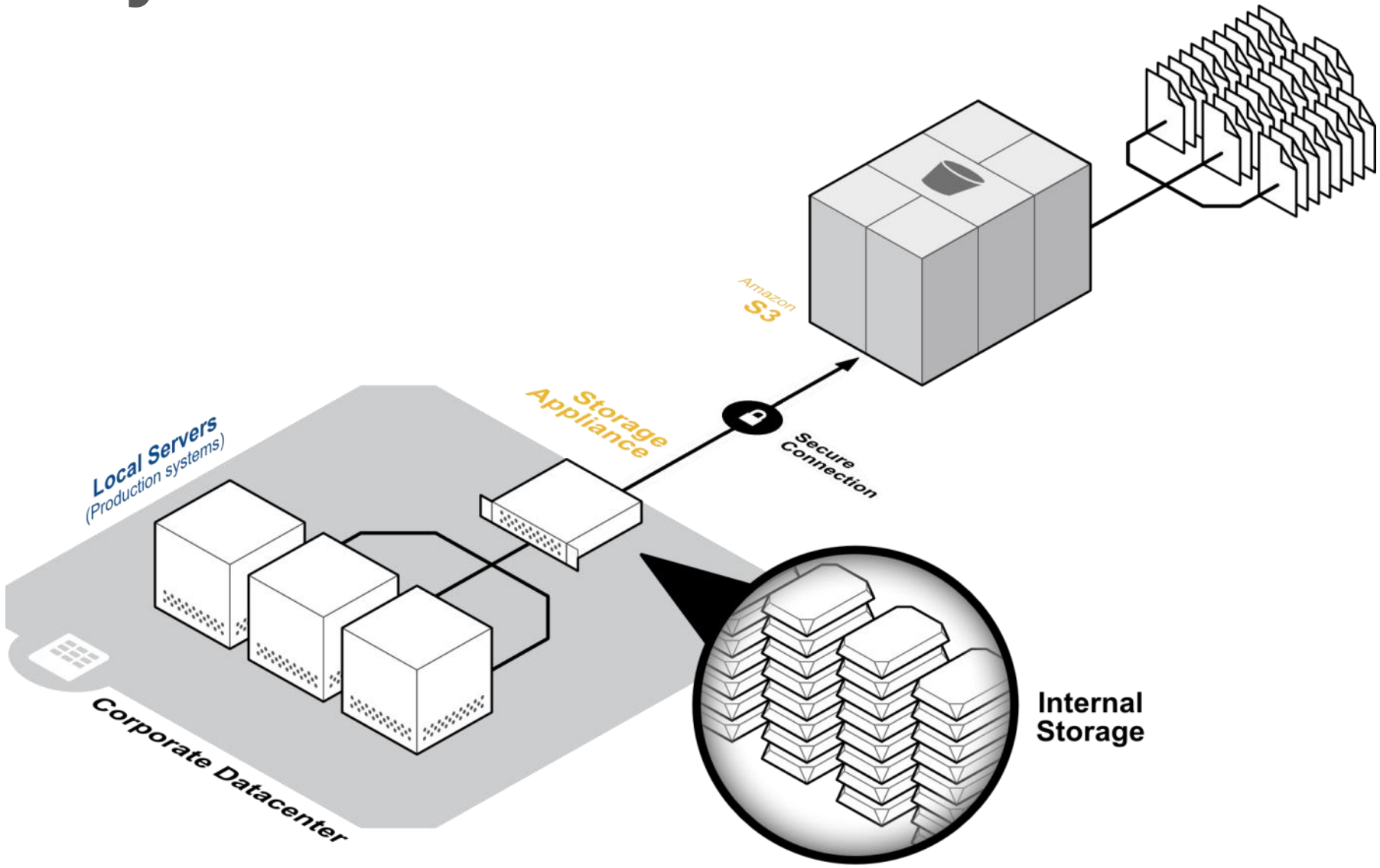


- EBS Magnetic
  - 40-200 IOPS
- EBS General Purpose
  - SSD backed
  - 3 IOPS / GB
  - Burstable to 3,000 IOPS
- EBS Provisioned IOPS
  - SSD backed
  - Up to 4,000 IOPS consistently

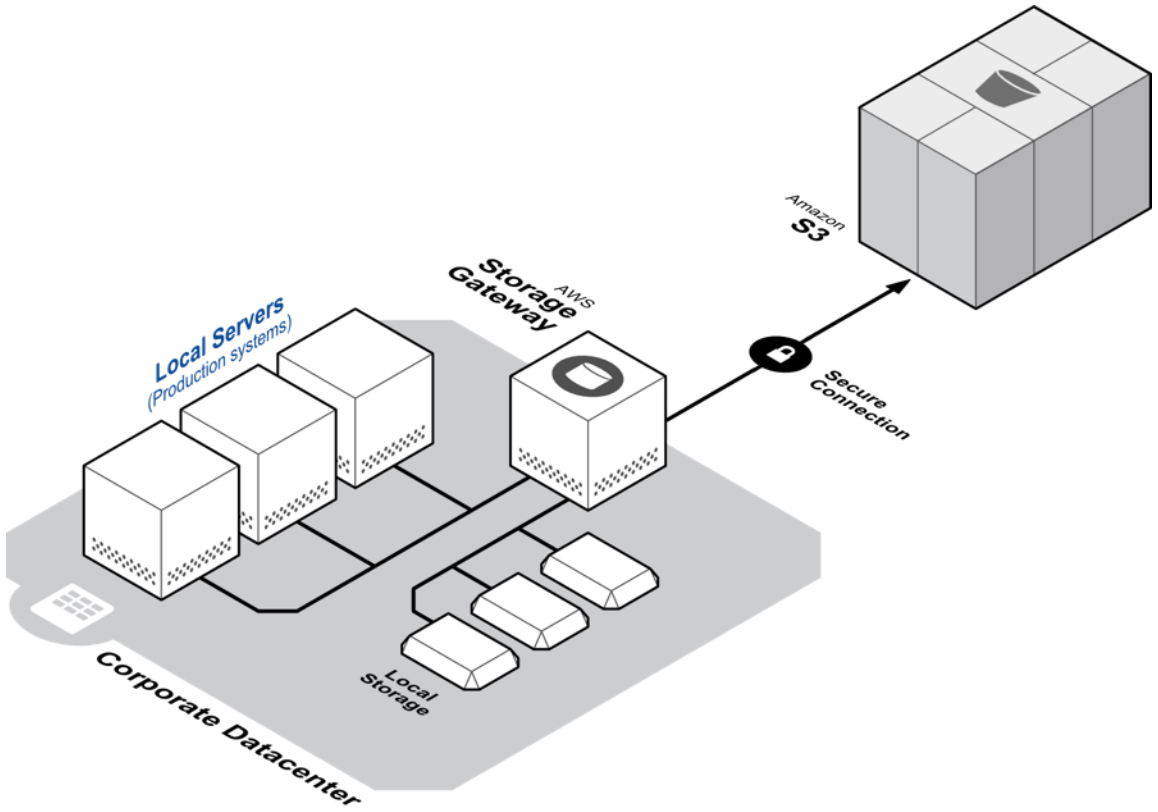
IOPS = I/O per second for up to 256KB blocks

Gateway Solution

Gateway Solution

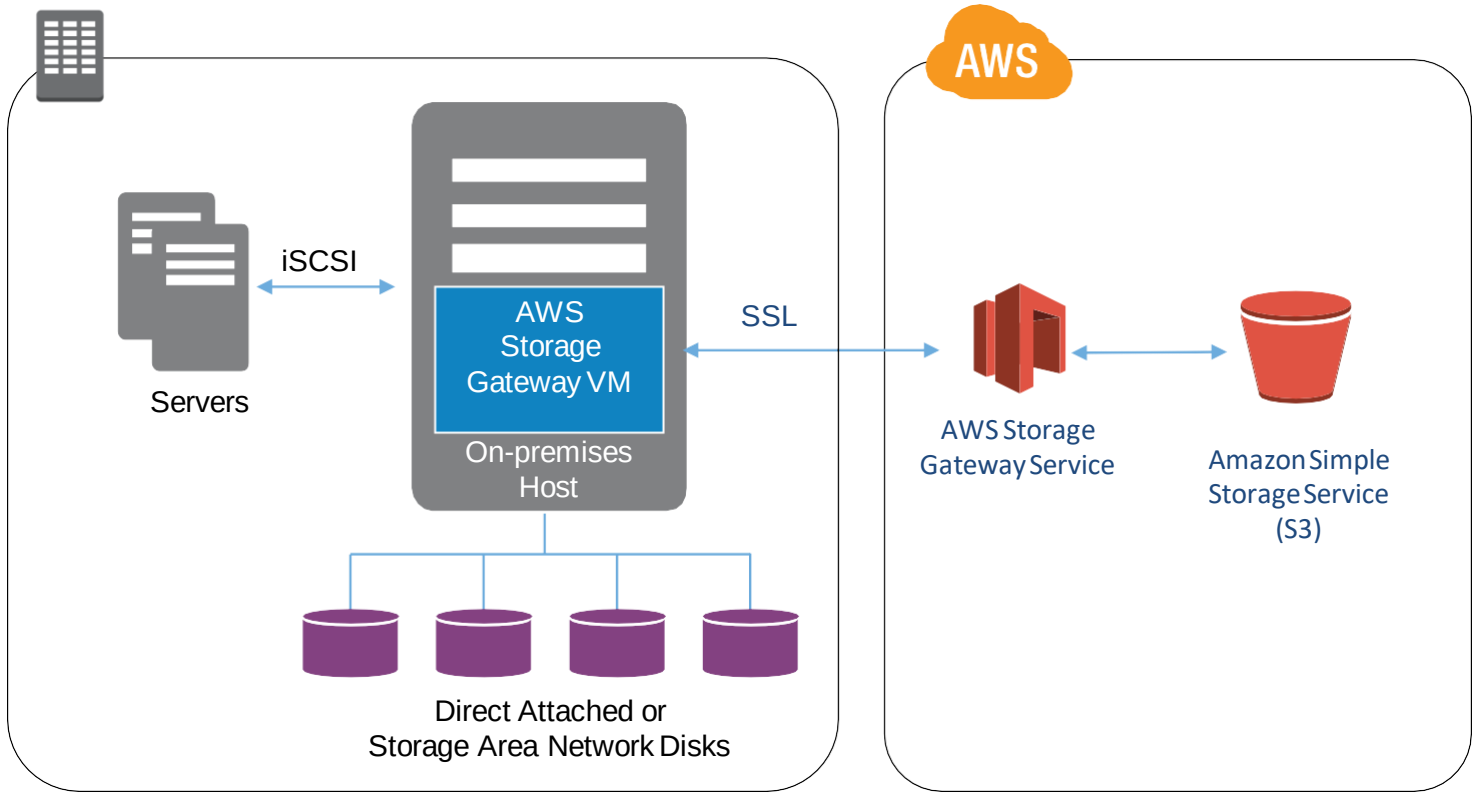


AWS Storage Gateway



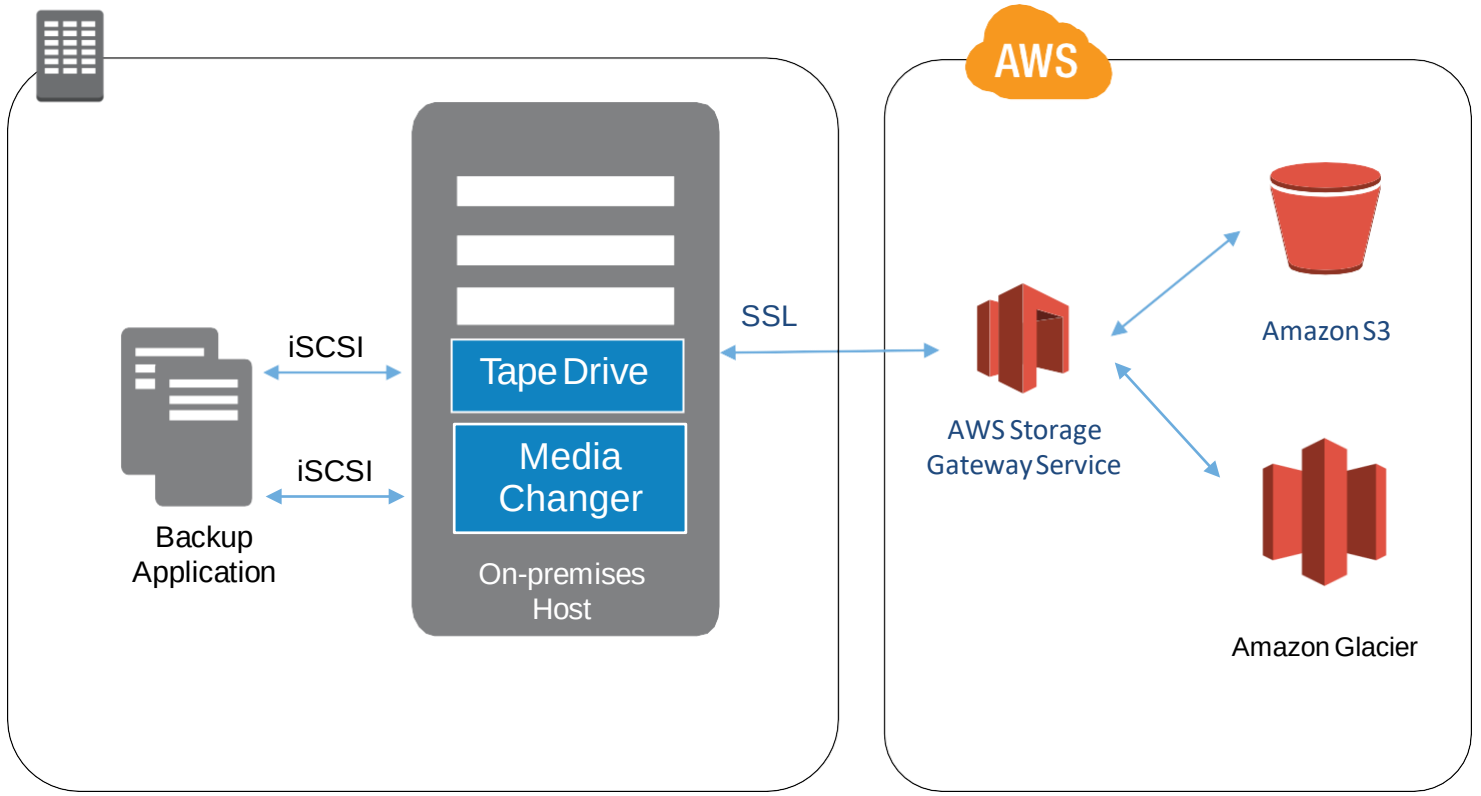
- Connect an on-premises software appliance to provide integration with Amazon S3
- Supports three configuration
  - Gateway-Cached Volumes
  - Gateway-Stored Volumes
  - Gateway-Virtual Tape Library (VTL)

AWS Storage Gateway



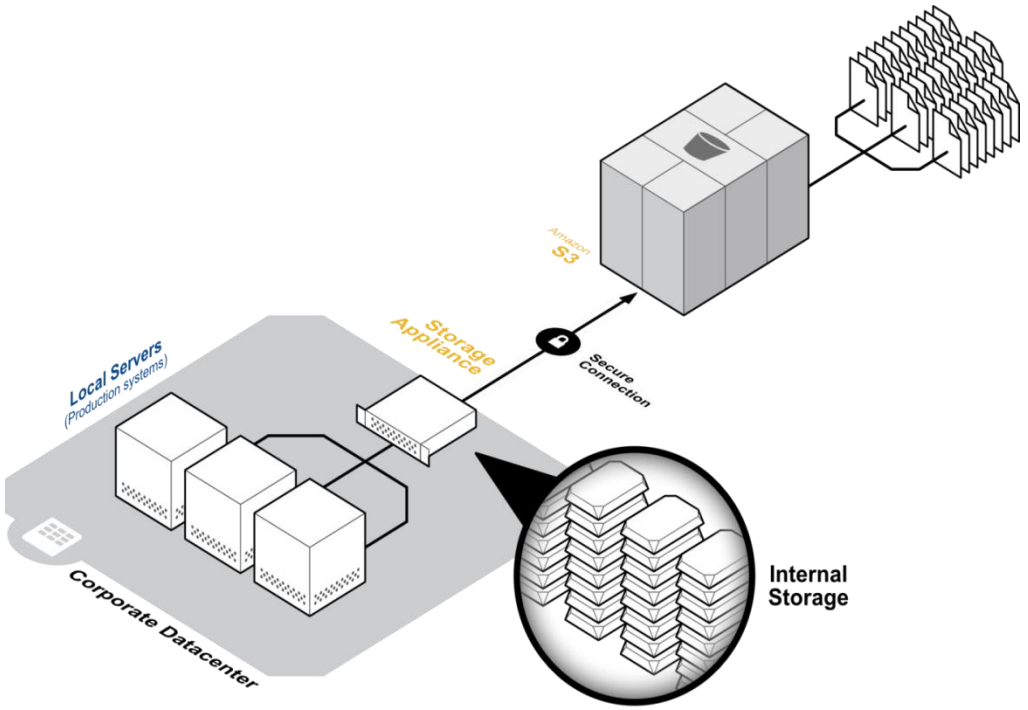
- Cached – Data is stored in Amazon S3 with frequently accessed files kept locally
- Stored – Asynchronous point-in-time backup snapshots to Amazon S3

AWS Storage Gateway



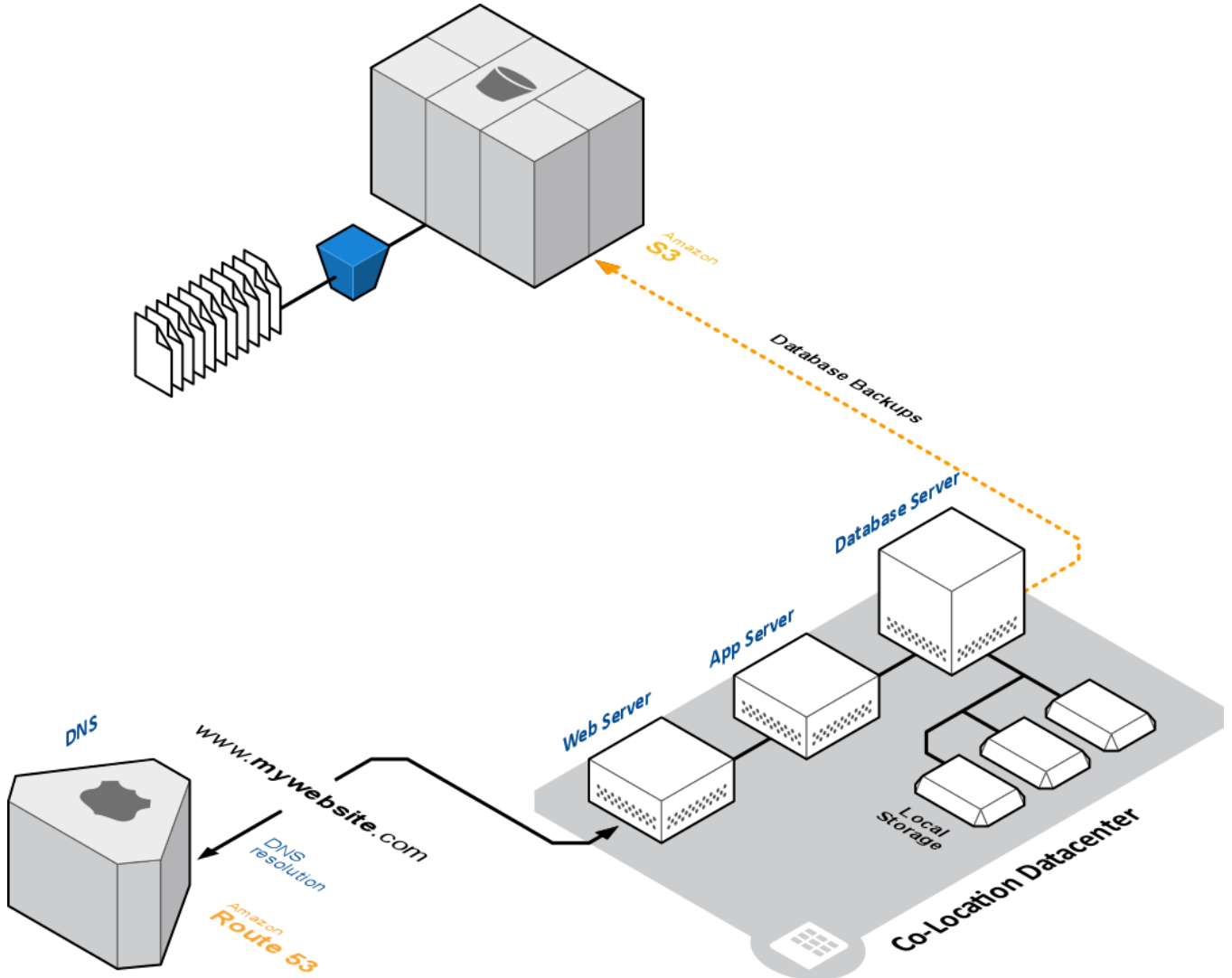
- VTL– Expose an industry standard virtual tape library
- Write to tape →→S3
- Put in shelf →→Glacier

AWS Storage Gateway

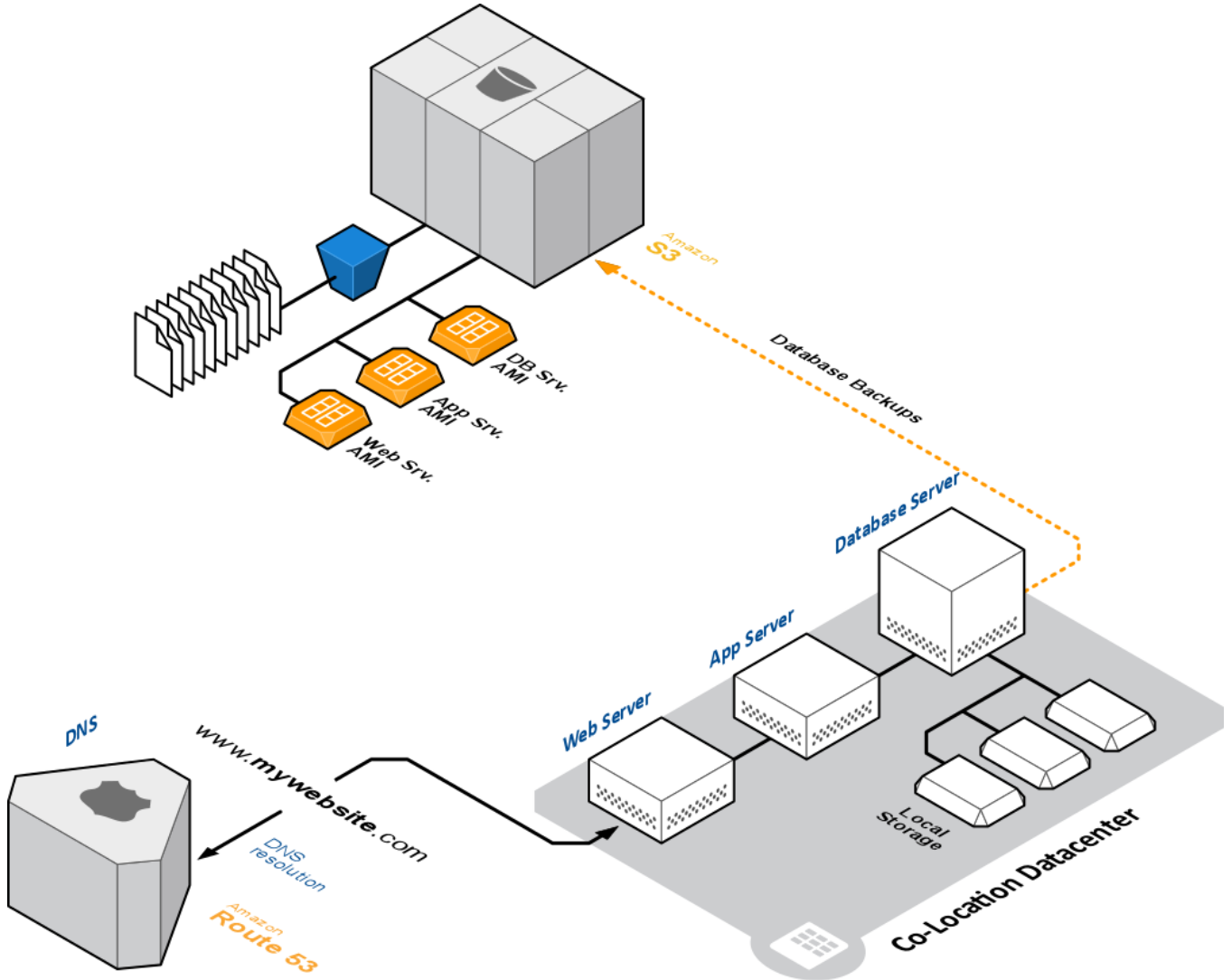


- Backup data into Amazon S3
- Disaster Recovery of applications to EC2
- Archive into Amazon Glacier

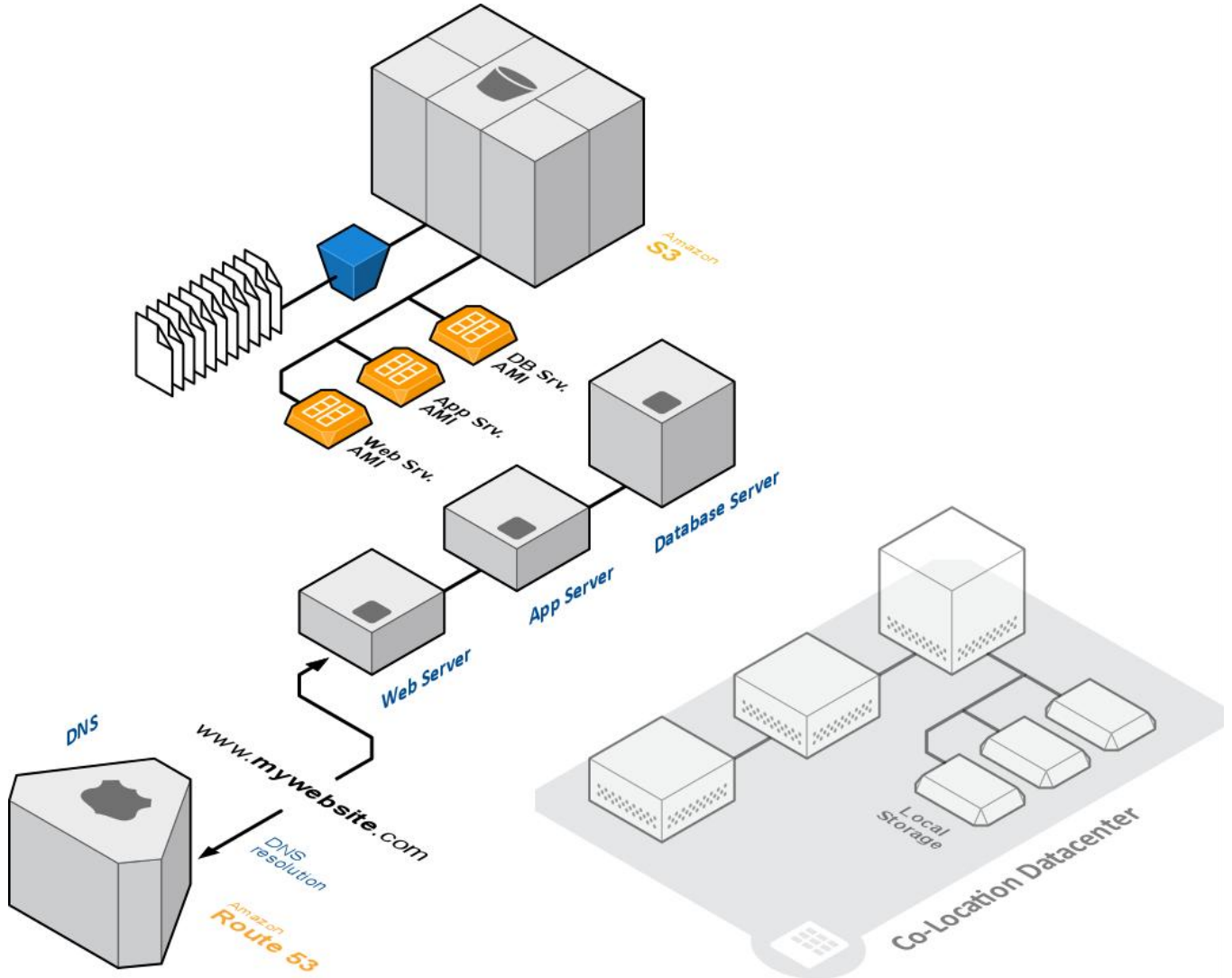
Use Case: Disaster Recovery



Use Case: Disaster Recovery



Use Case: Disaster Recovery



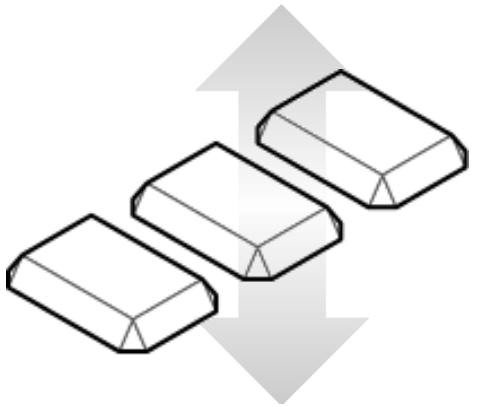
How to get my data into AWS?

Internet

Transfer data through secure encrypted tunnel over the public internet

AWS Direct Connect

Dedicated bandwidth between your site and AWS



AWS Import/Export

Physical transfer of media into and out of AWS

Why AWS for storage?

Reduce CAPEX while dramatically increasing scalability

Eliminate the need for secondary sites

Reduce costs

Eliminate on premise equipment to manage archives

Consolidate on-–premise and augment with cloud

Reduce on-–premise

Eliminate capacity planning

Eliminate provisioning for peak demand

Change processes

Remove tape archives

Cycle out aging disk arrays

Remove aging technologies

Demo Time!