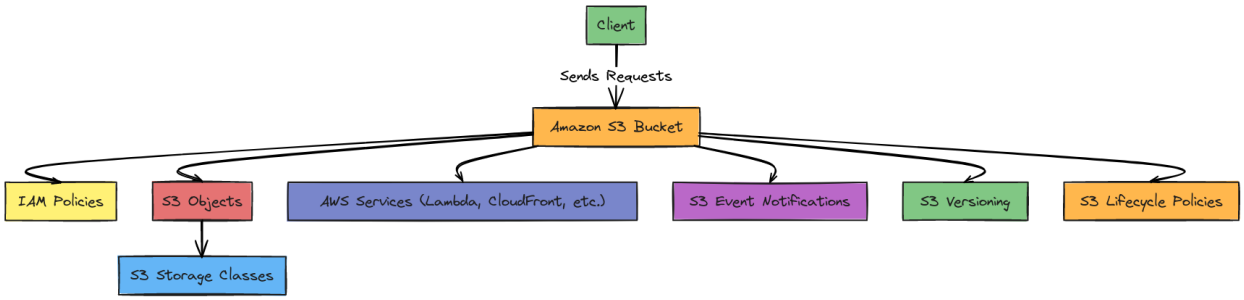
AWS S3 Buckets

Amazon S3

Amazon S3 is a scalable object storage service used to store and retrieve any amount of data, at any time, from anywhere.



Think of it as a giant, durable cloud-based file system — but for objects (not a typical file system like NTFS or ext4).

Concept	Description
Bucket	A container for storing objects (like folders in your computer)
Object	Files stored in buckets, including data, metadata, and key (name)
Key	Unique identifier for an object within a bucket
Region	Buckets are created in specific AWS regions for latency and compliance
Storage Class	Defines the availability and pricing tier (Standard, Glacier, etc.)

Bucket Structure

s3://my-bucket-name/

```
├── image1.jpg
├── folder1/
│   └── doc.pdf
└── folder2/
    └── logs.txt
```

Objects are stored in a flat namespace with "prefixes" simulating folder-like structures.

S3 Storage Classes:

Storage Class	Use Case	Durability	Availability	Cost
Standard	Frequently accessed data	99.999999999%	99.99%	High
Intelligent-Tiering	Automatically moves data between tiers	Same as Standard	Same	Medium
Standard-IA	Infrequent access but fast retrieval	Same	99.9%	Lower
One Zone-IA	Infrequent access, 1 AZ only	Lower	Lower	Lower
Glacier Instant	Archiving with instant access	High	Lower	Cheap
Glacier Flexible	Archiving with minutes to hours delay	High	Lower	Very Cheap
Glacier Deep Archive	Long-term archive, up to 12 hours	High	Lower	Cheapest

Security Features

Feature	Purpose
Bucket Policies	JSON-based access control for buckets
IAM Policies	User/role-based access to buckets/objects
ACLs (Access Control Lists)	Legacy method (not preferred)
Encryption	Server-side (SSE-S3, SSE-KMS, SSE-C) or client-side
Block Public Access	Prevents accidental public exposure

Accessing S3

Console: Web interface

CLI/SDK: Powerful programmatic access

URL Access:

```
https://<bucket-name>.s3.<region>.amazonaws.com/<object-key>
```

Hands-On: Uploading File to S3

Step 1: Create a Bucket

```bash
aws s3api create-bucket \
  --bucket my-learning-bucket \
  --region ap-south-1 \
  --create-bucket-configuration LocationConstraint=ap-south-1
```

Step 2: Upload a File

```bash
aws s3 cp myfile.txt s3://my-learning-bucket/
```

Step 3: Make it Public (⚠ Only for testing)

```bash
aws s3api put-object-acl \
  --bucket my-learning-bucket \
  --key myfile.txt \
  --acl public-read
```

Step 4: Download/Access via URL

```
https://my-learning-bucket.s3.ap-south-1.amazonaws.com/myfile.txt
```

Lifecycle Management

Lifecycle rules can:

- **Move data to cheaper storage tiers**
- **Expire/delete objects after X days**
- **Transition to Glacier after inactivity**

Example rule:

```
json
{
  "ID": "ArchiveRule",
  "Prefix": "logs/",
```

```
  "Status": "Enabled",
  "Transitions": [{
    "Days": 30,
    "StorageClass": "GLACIER"
  }]
}
```

S3 Versioning

Keep multiple versions of an object

Protect against accidental deletion/overwrite

Once enabled, can’t be disabled (only suspended)

Real-World Use Cases

Use Case	Why S3?
Website hosting	Static site hosting supported
Backup & archival	Durability and cost-effective tiers
Data lake for analytics	Integrates with Athena, Redshift
Media storage	Scalable and secure
Logs & monitoring	Centralized, versioned storage

Limitations / Considerations:

Limitation	Note
5 TB max object size	Multipart upload required for large files
100 bucket limit (soft)	Can be increased with support
Flat hierarchy	No real folders, just prefixes
No append operations	Need to rewrite object

Best Practices

- Enable versioning
- Use SSE-KMS for encryption
- Apply bucket policies with least privilege
- Enable MFA Delete
- Set up logging and access monitoring