Content Delivery Network

- A Content delivery network, or content distribution network (CDN) is a geographically distributed network of proxy servers and their data centers.

- - It is geographically distributed group of servers that would work together to provide fast delivery of Internet content.

Content Delivery Network

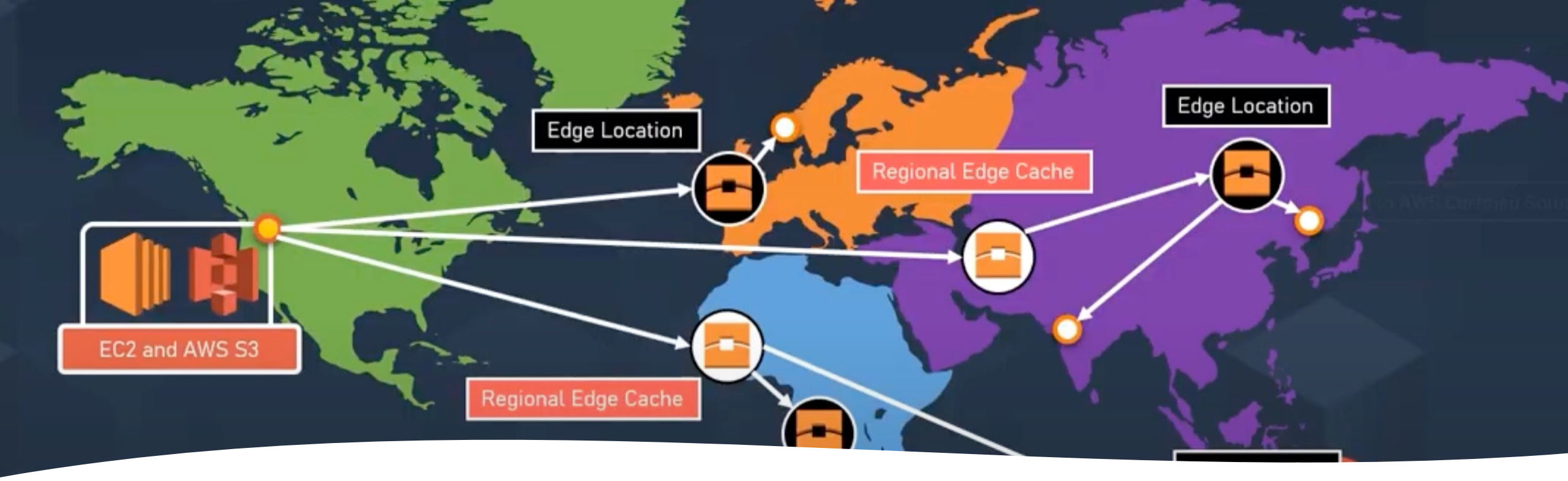
- **Placing datacenters Closer**
- **High/Global Availability**
- **Content Caching at the Locations**
- **Static Content, High Read Performance**
- **Reducing bandwidth costs**
- **Improving page load times**

Amazon CloudFront

Fast, Highly secure and programmable content delivery network (CDN)

- Amazon CloudFront is a fast content delivery network (CDN) service that securely delivers data, videos, application and APIs to customer globally.
- CloudFront is integrated with AWS – both physical locations that are directly connected to the AWS global infrastructure.
- AWS Shield for DDoS mitigation.
- Amazon S3
- Elastic Load Balancing or Amazon EC2 as origins for your applications.
- Lambda@Edge to run custom code
- If you use AWS origins such as Amazon S3, Amazon Ec2 or Elastic Load Balancing. You do not pay for any data transferred between these services and cloudFront.
- Integrates with AWS Web Application Firewall.
- You can use CloudFront with APIs, AWS Management Console, AWS CloudFormation, CLIs and SDKs.
- Amazon CloudFront can be used to secure and accelerate your webSocket traffic as well as API calls.
- CloudFront supports proxy methods(PUT,POST,OPTIONS, DELETE, PATCH) and is already integrated with Amazon API gateway by default.
- Allows you to communicate with external HTTPS and talk to internal HTTPS backends.
- Usage and Support:
    - Pay-as-you-go pricing model with no upfront fees
    - No required long-term contracts.
    - Support for the CDN is included in your existing AWS support subscription.

To deliver content to end users with lower latency. Amazon CloudFront uses a global network of 216 Points of Presence (205 Edge Locations and 11 Regional Caches) in 84 cities across 42 countries.



Edge Location

Edge location is where end-users access services located in aws , so they are located in most of the major cities around the world and specifically used by cloudFront as a CDN to distribute content to end users to reduce latency.

OR

You can think , edge location is a collection of physical servers within a data centers that can help you access the aws services so that your users don’t have to spend more time accessing the same resource located thousands of miles apart from their house but from the place they are actually trying to access it.

Regional Edge Cache & Edge Location

An edge location is the location where content is cached (separate to AWS regions/AZs).

Requests are automatically routed to the nearest edge location.

Edge locations are not tied to Availability Zones or regions.

Regional Edge Cache are located between origin web servers and global edge locations and have a larger cache.

Regional Edge Caches have larger cache-width than any individual edge location, so your objects remain in cache longer at these locations.

Regional Edge caches aim to get content closer to users.

Objects are cached for 24 hours by default.

AWS CloudFront - Origins

What are the origins that you can set for CloudFront to serve your Data?

An origin is the location where content is stored, and from which CloudFront gets content to serve to viewers.Origin can be either an S3 bucket,an Ec2 instance , an Elastic Load Balancer or Route 53 - can also be external (non- aws).

S3OriginConfig:

Use this type to specify an Amazon S3 bucket that is not configured with static website hosting.

CustomOriginConfig:

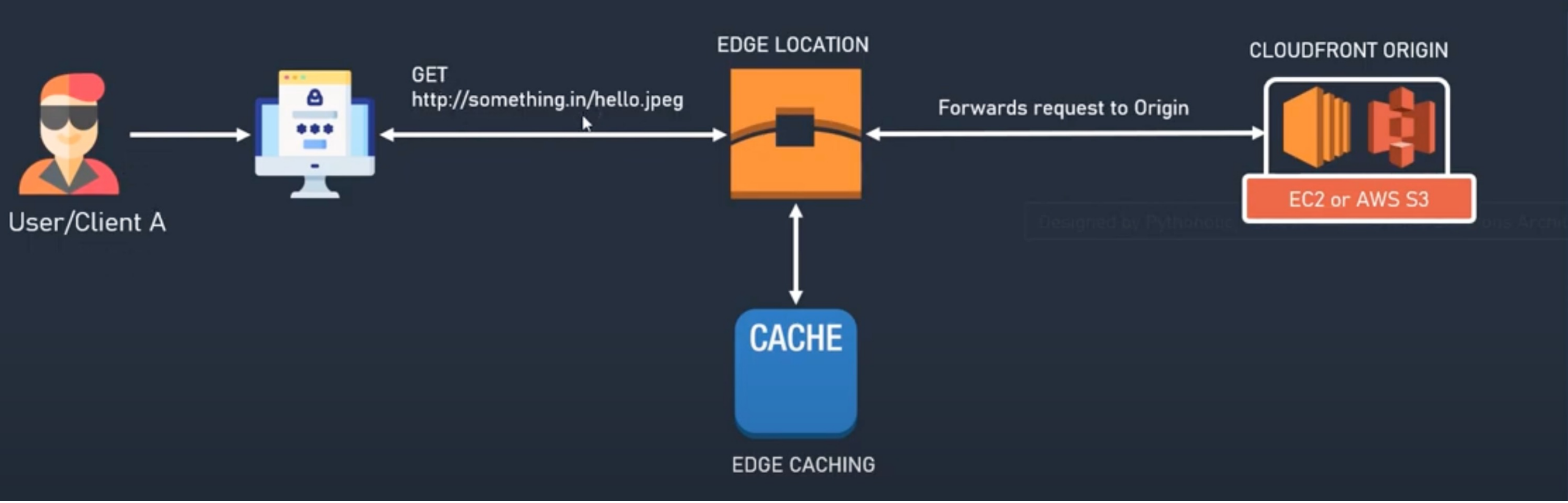
Use this type to specify various other kinds of content containers or HTTP Servers.

- Amazon S3 bucket that is configured static website hosting
- Elastic Load Balancing load balancer
- AWS Elemental MediaStore container
- HTTP server, running on an Amazon Ec2 instance or any other kind of host

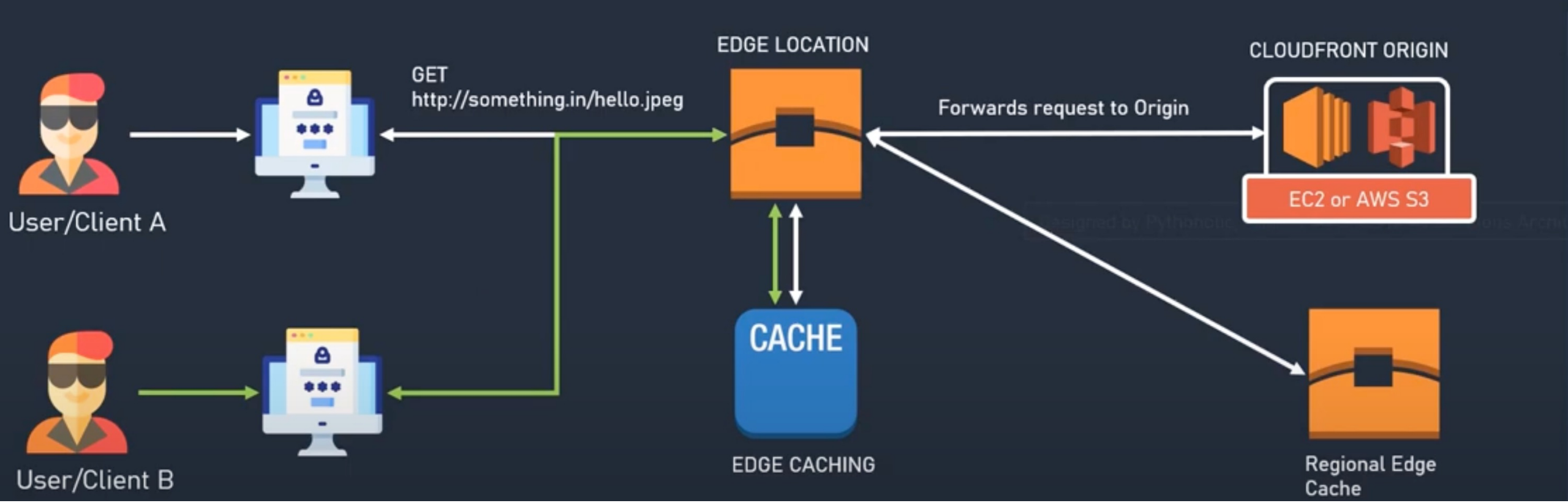
Pricing

Free Tier

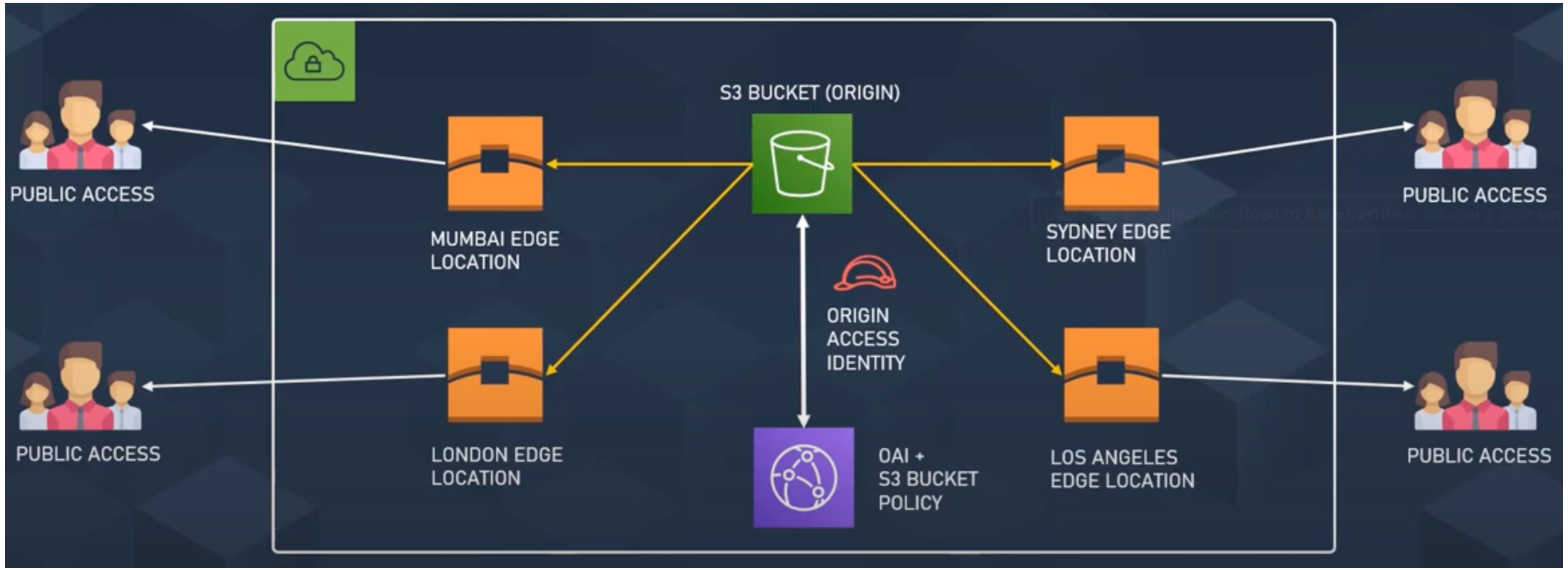
- **50 GB OF DATA TRANSFER OUT – 12 MONTH FREE**
- **2,000,000 HTTP OR HTTPS REQUESTS – EACH MONTH FOR ONE YEAR**



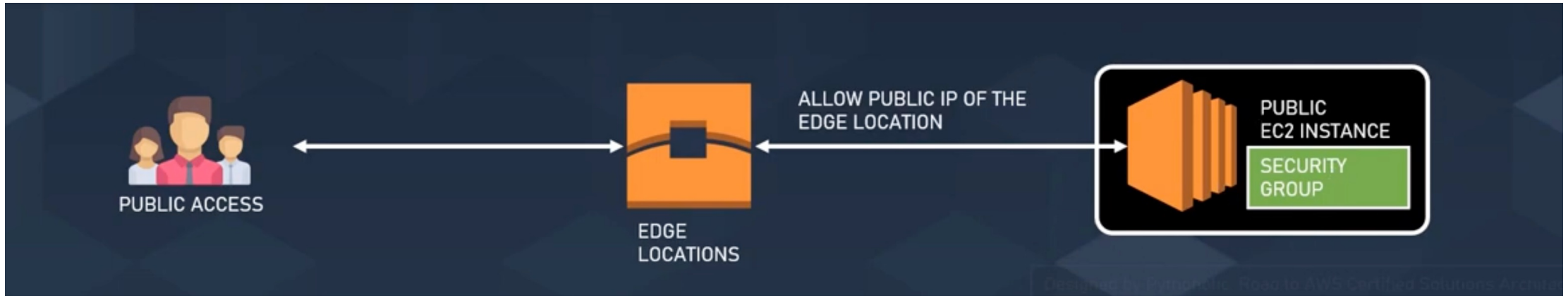
How Does Cloud Front work and serve requests?



How Does Cloud Front work and serve requests?



Restricting Access to Amazon S3 Content by using an Origin Access Identity



Using EC2 Instance as Origin for Cloud Front

Using ALB as Origin for CloudFront

How can I restrict users in certain locations from accessing web content served by my CloudFront distribution?

- To use geo restriction, you have two options:
  - Use the CloudFront geo restriction
  - Use a third-party geolocation service

- Use the CloudFront geo restriction:
  - WHITELIST: Allow your users to access your content only if they're in one of the countries on a whitelist of approve.
  - BLACKLIST: Prevent your users from accessing your content if they are in one of the countries on a blacklist of banned countries.

  - Process:
  - You add a whitelist that contains only the name of the location
  - When you send a request, DNS routes the request to the CloudFront edge location
  - Edge location determines that user is not allowed or not to download your content.
  - CloudFront returns an HTTP status code 403 (Forbidden) to the user. (If the user is not allowed)

- Use a third-party geolocation service:
  - Combine CloudFront with a third-party geolocation service.
  - With this you can restrict based on city, zip or postal code, or even latitude and longitude
  - When using 3rd party Geo Location, it is recommended that you use CloudFront signed URLs

Distributions

To distribute content with CloudFront you need to create a distribution.

The distribution includes the configuration of the CDN including:

- Content Origins.
- Access (public or restricted)
- Security (HTTP or HTTPS)
- Cookie or query-string forwarding.
- Access logs (record viewer activity)

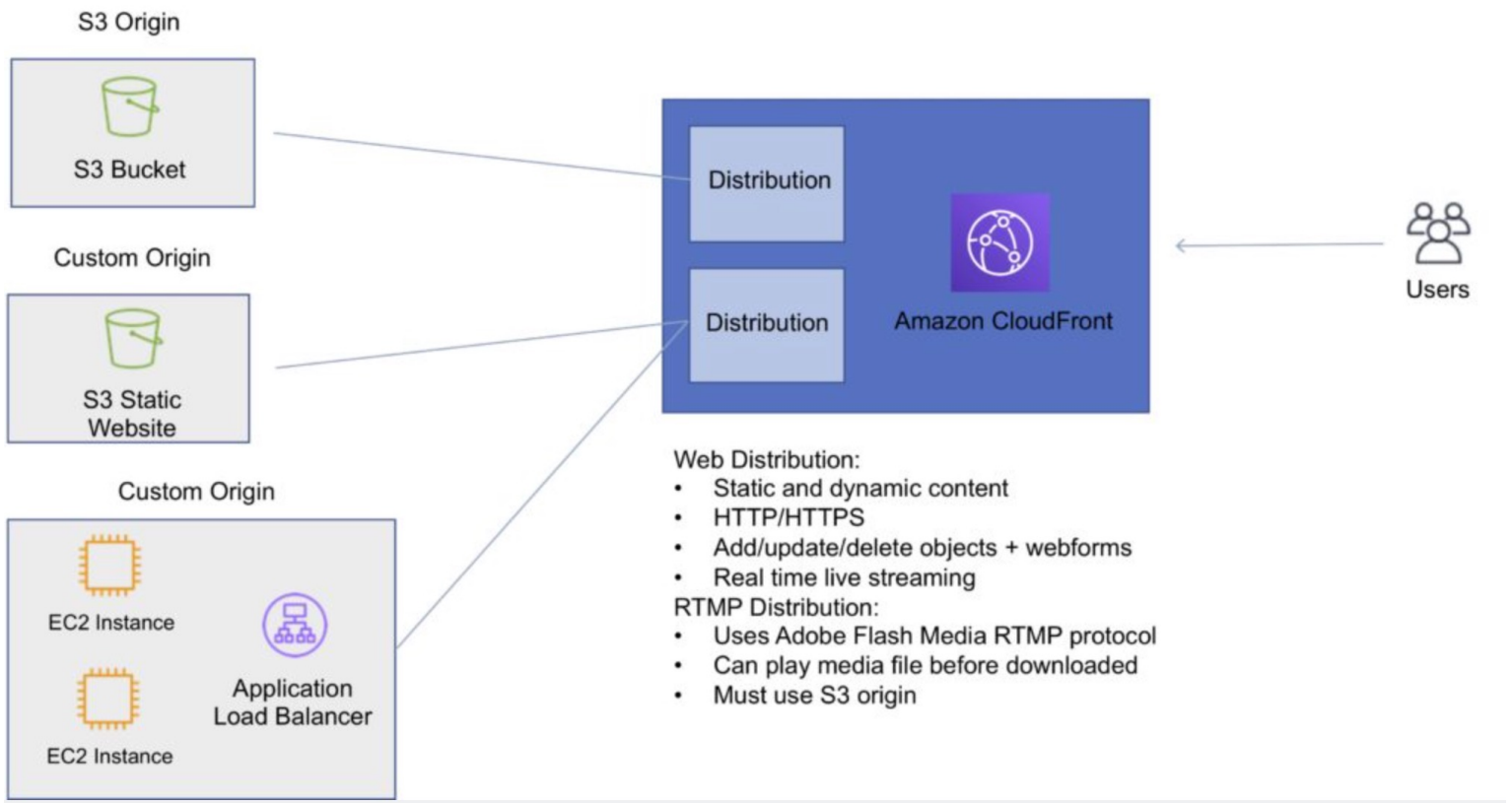
Types of Distribution

Web Distribution:

- Static and dynamic content including .html, .css, .php, and graphics files.
- Distributes files over HTTP and HTTPS.
- Add, update, or delete objects, and submit data from web forms.
- Use live streaming to stream an event in real time.

RTMP:

- Distribute streaming media files using Adobe Flash Media Server's RTMP protocol.
- Allows an end user to begin playing a media file before the file has finished downloading from a CloudFront edge location.
- Files must be stored in an S3 bucket.



Depiction of Amazon CloudFront Distributions and Origins

Cache Behavior

Allows you to configure a variety of CloudFront functionality for a given URL path pattern.

For each cache behavior you can configure the following functionality:

- The path pattern (e.g. /images/*.jpg, /images*.php).
- The origin to forward requests to (if there are multiple origins).
- Whether to forward query strings.
- Whether to require signed URLs.
- Allowed HTTP methods.
- Minimum amount of time to retain the files in the CloudFront cache (regardless of the values of any cache-control headers).

The default cache behavior only allows a path pattern of /*.

Additional cache behaviors need to be defined to change the path pattern following creation of the distribution.

You can restrict access to content using the following methods:

- Restrict access to content using signed cookies or signed URLs.
- Restrict access to objects in your S3 bucket.

Cache Behavior

A special type of user called an Origin Access Identity (OAI) can be used to restrict access to content in an Amazon S3 bucket.

By using an OAI you can restrict users so they cannot access the content directly using the S3 URL, they must connect via CloudFront.

You can define the viewer protocol policy:

- HTTP and HTTPS.
- Redirect HTTP to HTTPS.
- HTTPS only.

You can define the Allowed HTTP Methods:

- GET, HEAD.
- GET, HEAD, OPTIONS.
- GET, HEAD, OPTIONS, PUT, POST, PATCH, DELETE.

For web distributions you can configure CloudFront to require that viewers use HTTPS.

Cache Behavior

Field-Level Encryption:

- Field-level encryption adds an additional layer of security on top of HTTPS that lets you protect specific data so that it is only visible to specific applications.
- Field-level encryption allows you to securely upload user-submitted sensitive information to your web servers.
- The sensitive information is encrypted at the edge closer to the user and remains encrypted throughout application processing.

Origin policy:

- HTTPS only.
- Match viewer – CloudFront matches the protocol with your custom origin.
- Use match viewer only if you specify Redirect HTTP to HTTPS or HTTPS only for the viewer protocol policy.
- CloudFront caches the object once even if viewers make requests using HTTP and HTTPS.

Object invalidation:

- You can remove an object from the cache by invalidating the object.
- You cannot cancel an invalidation after submission.
- You cannot invalidate media files in the Microsoft Smooth Streaming format when you have enabled Smooth Streaming for the corresponding cache behaviour.

Cache Behavior

- Objects are cached for the TTL (always recorded in seconds, default is 24 hours, default max is 1 year).
- Only caches for GET requests (not PUT, POST, PATCH, DELETE).
- Dynamic content is cached.

Consider how often your files change when setting the TTL.

- Invalidation can be used to immediately revoke cached objects – chargeable.

Cache Hit Ratio

A good cache hit ratio means more requests are served from the cache.

Methods of improving the cache hit ratio include:

- Use the Cache-Control max-age directive to increase the time objects remain in the cache
- Use Origin Shield.
- Forward only the query string parameters for which your origin will return unique objects.
- Configure CloudFront to forward only specified cookies instead of forwarding all cookies.
- Configure CloudFront to forward and cache based on only specified headers instead of forwarding and caching based on all headers.

Lambda@Edge

Can be used to run Lambda at Edge Locations.

Let's you run Node.js and Python Lambda functions to customize content that CloudFront delivers.
Executes the functions in AWS locations closer to the viewer.

You can use Lambda functions to change CloudFront requests and responses at the following points:

- After CloudFront receives a request from a viewer (viewer request).
- Before CloudFront forwards the request to the origin (origin request).
- After CloudFront receives the response from the origin (origin response).
- Before CloudFront forwards the response to the viewer (viewer response).

Lambda@Edge can do the following:

- Inspect cookies and rewrite URLs to perform A/B testing.
- Send specific objects to your users based on the User-Agent header.
- Implement access control by looking for specific headers before passing requests to the origin.
- Add, drop, or modify headers to direct users to different cached objects.
- Generate new HTTP responses.
- Cleanly support legacy URLs.
- Modify or condense headers or URLs to improve cache utilization.
- Make HTTP requests to other Internet resources and use the results to customize responses.

Signed URLs and Signed Cookies

A signed URL includes additional information, for example, an expiration date and time, that gives you more control over access to your content. This additional information appears in a policy statement, which is based on either a canned policy or a custom policy.

CloudFront signed cookies allow you to control who can access your content when you don't want to change your current URLs or when you want to provide access to multiple restricted files, for example, all the files in the subscribers' area of a website.

Application must authenticate user and then send three Set-Cookie headers to the viewer; the viewer stores the name-value pair and adds them to the request in a Cookie header when requesting access to content.

Use signed URLs in the following cases:

- You want to restrict access to individual files, for example, an installation download for your application.
- Your users are using a client (for example, a custom HTTP client) that doesn't support cookies.

Use signed cookies in the following cases:

- You want to provide access to multiple restricted files, for example, all the files for a video in HLS format or all the files in the subscribers' area of website.
- You don't want to change your current URLs.

Origin Access Identity

- Used in combination with signed URLs and signed cookies to restrict direct access to an S3 bucket (prevents bypassing the CloudFront controls).
- An origin access identity (OAI) is a special CloudFront user that is associated with the distribution.
- Permissions must then be changed on the Amazon S3 bucket to restrict access to the OAI.
- If users request files directly by using Amazon S3 URLs, they're denied access.
- The origin access identity has permission to access files in your Amazon S3 bucket, but users don't.

Charges

There is an option for reserved capacity over 12 months or longer (starts at 10TB of data transfer in a single region).

You pay for:

- Data Transfer Out to Internet.
- Data Transfer Out to Origin.
- Number of HTTP/HTTPS Requests.
- Invalidation Requests.
- Dedicated IP Custom SSL.
- Field level encryption requests.

You do not pay for:

- Data transfer between AWS regions and CloudFront.
- Regional edge cache.
- AWS ACM SSL/TLS certificates.
- Shared CloudFront certificates.


THANK
YOU