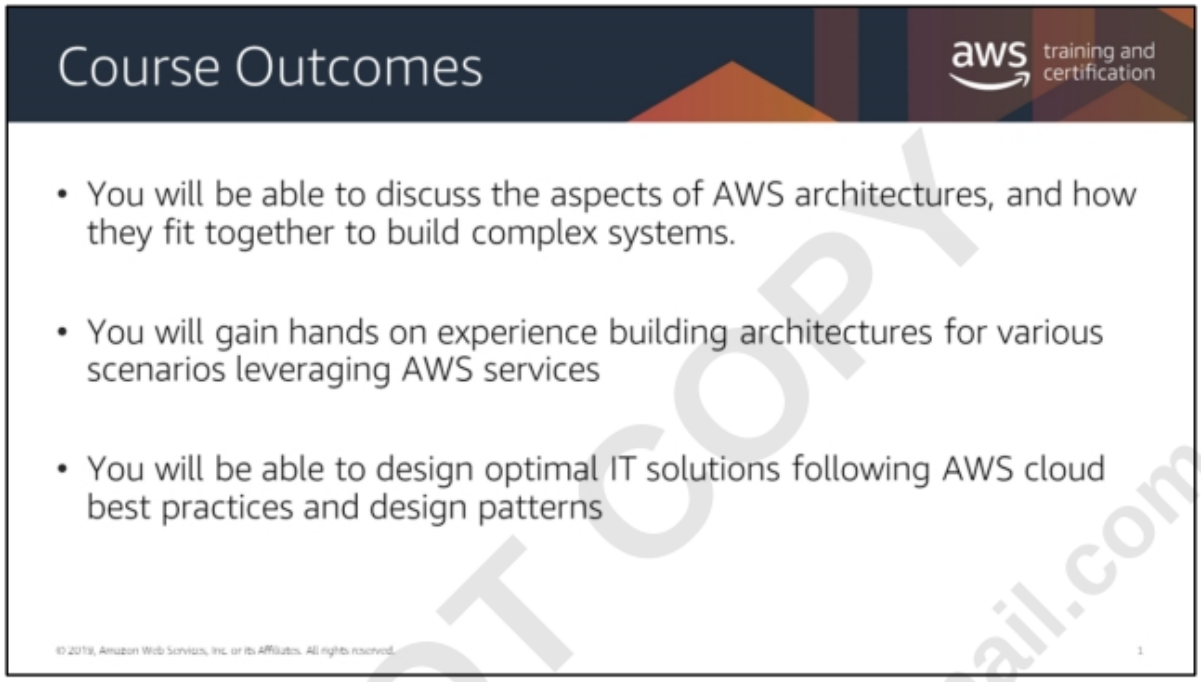
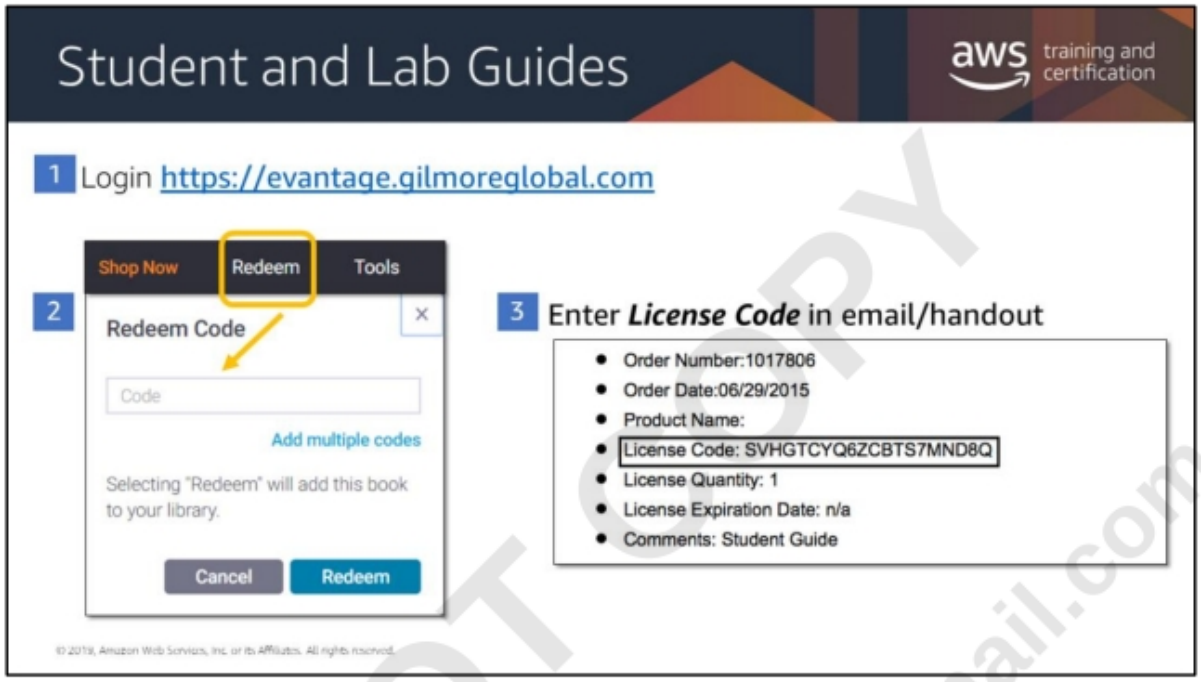
Corrections or feedback on the course, please email us at:
aws-course-feedback@amazon.com.
For all other questions, contact us at:
https://aws.amazon.com/contact-us/aws-training/.

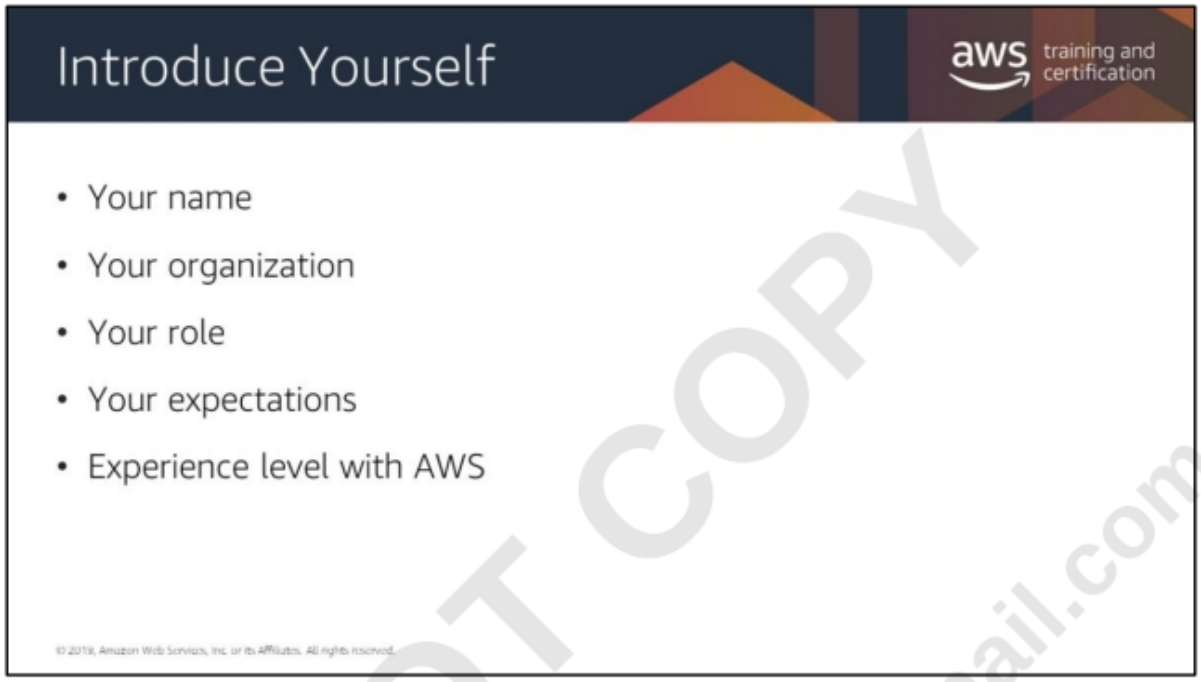
Contents

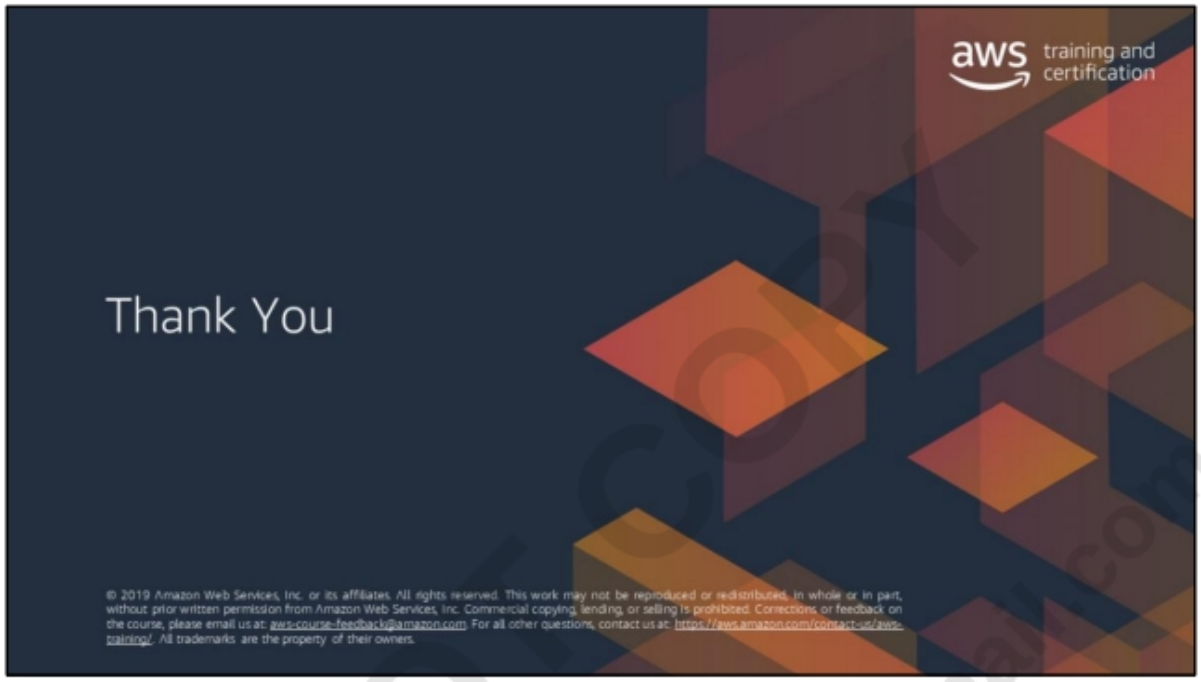
/





To login to access your student and lab guides, see: http://online.vitalsource.com





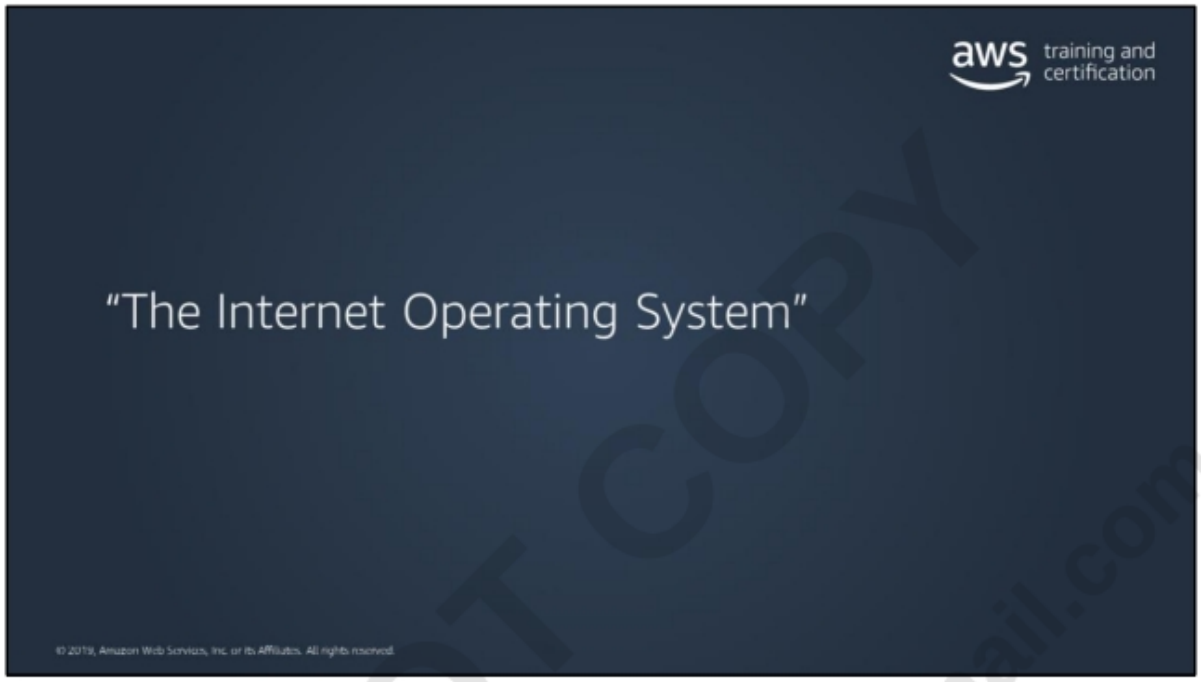
/

What's Coming Up?

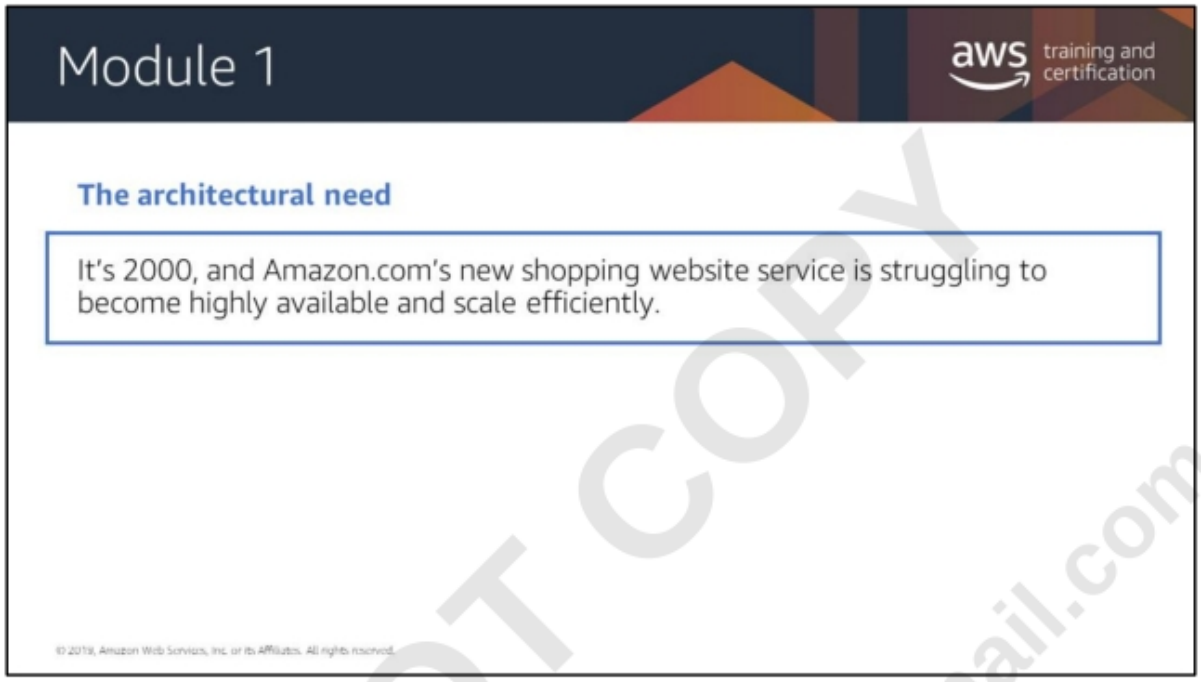
aws training and certification

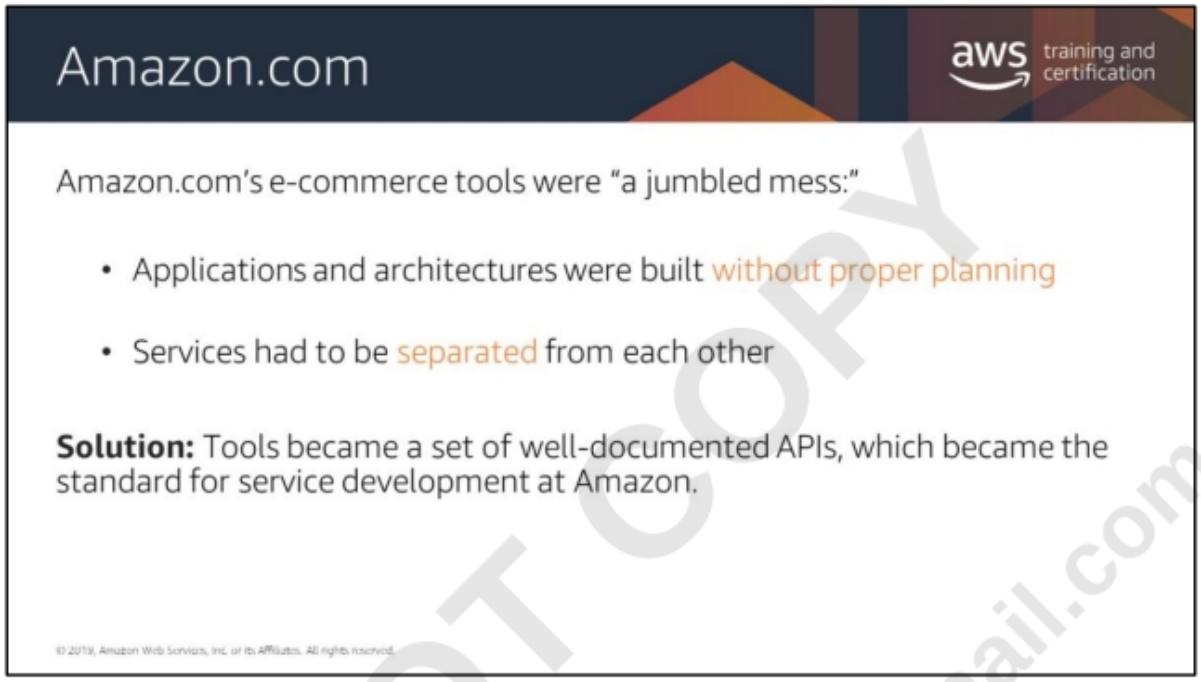
A quick review:

- What is the cloud? What is AWS?
- Design guidelines of the cloud
- The Well-Architected Framework
- AWS global infrastructure
- Large-scale architectural design



/





https://techcrunch.com/2016/07/02/andy-jassys-brief-history-of-the-genesis-of-aws/

/

Problems Persisted

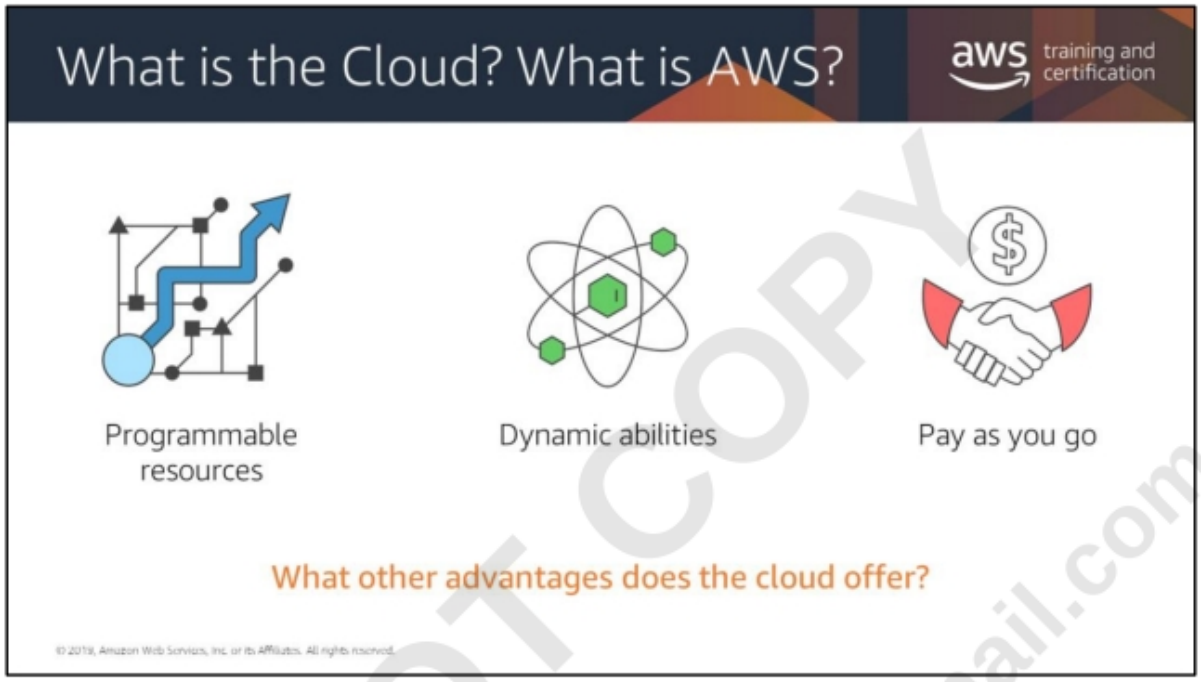
aws training and certification

Amazon.com still struggled to build applications quickly.

- Database, compute, and storage components took 3 months to build.
- Each team built their own resources, with no planning for scale or re-usability.

Solution: Built internal services to create highly available, scalable, and reliable architectures on top of their infrastructure. In 2006, started selling these services as AWS.

/



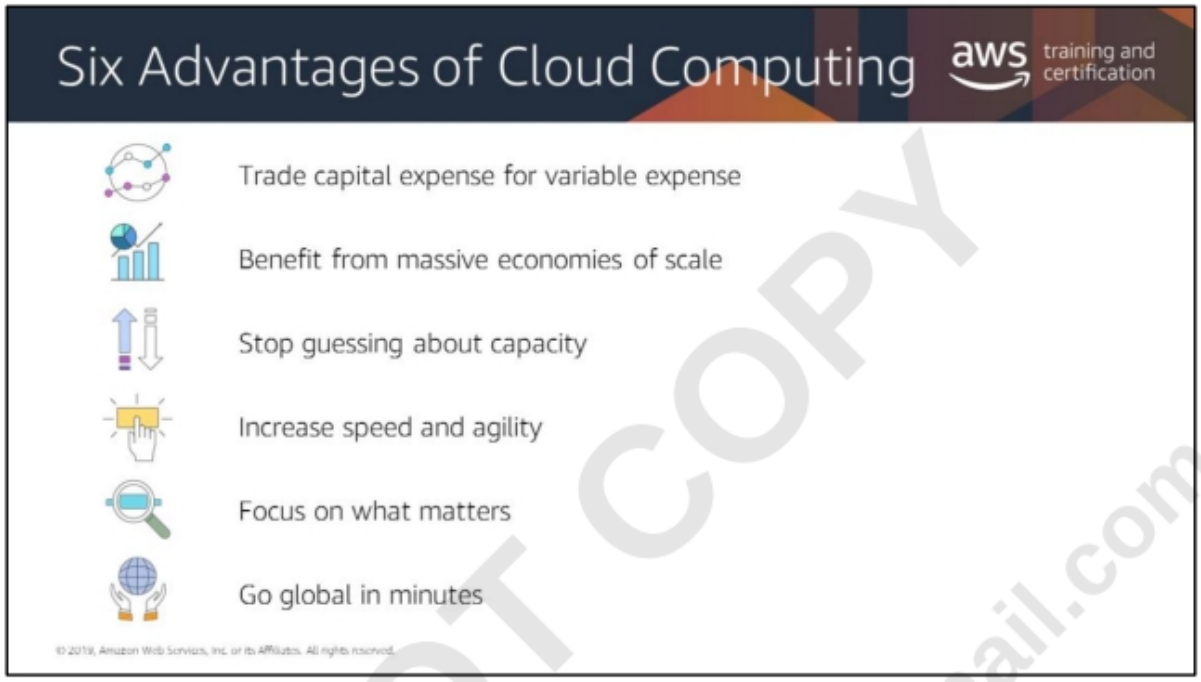
The cloud gives enormous advantage to those who can leverage its unique powers. The use of IT assets as programmatic resources allows you quickly set up and tear down infrastructure in a way that isn't possible with a traditional approach.

Access to these resources allows you to move forward in a very dynamic fashion. You can increase your database throughput or compute power with just a few clicks of the mouse. This provides an agility and flexibility that can really make the difference to your business.

Additionally, one of the biggest benefits of cloud computing is the ability to pay as you go. Letting you test and leverage the system without being fully committed. You can stop using these services at any time and change tactics to fit your needs.

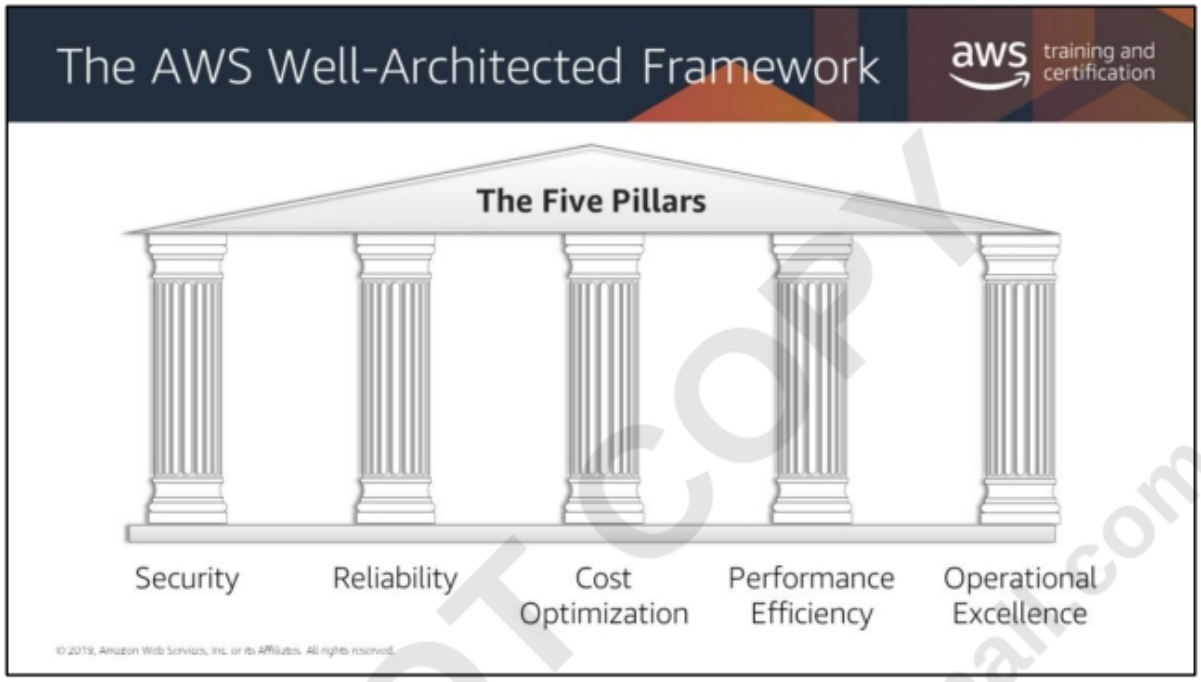
Let's talk about the six advantages of cloud computing with AWS. For more information, see https://aws.amazon.com/what-is-cloud-computing

/



For more information about the six primary benefits of cloud computing with AWS, see
https://www.youtube.com/watch?v=yMJ75k9X5_8

/



First we'll talk about some of the goals of the design principles in the Well-Architected Framework.

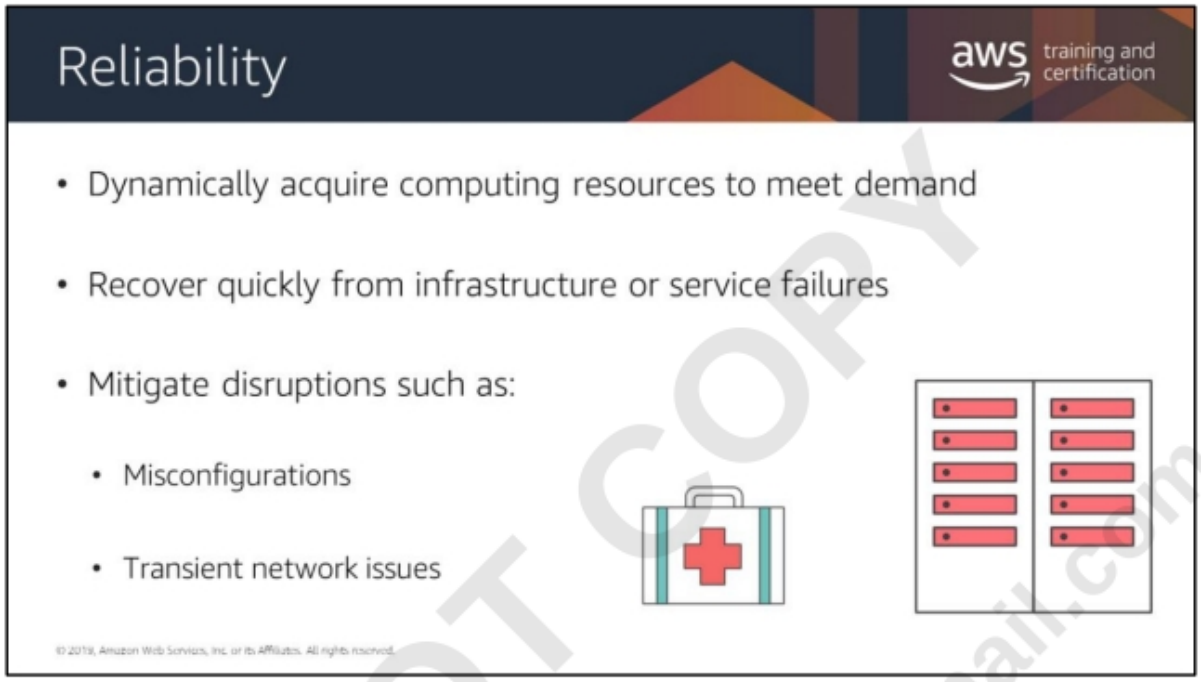
If you would like to have some assistance with well architected design:

The AWS Well-Architected Tool is a self-service tool that provides you with on-demand access to current AWS best practices. It is designed to help architects and their managers review AWS workloads at any time, without the need for an AWS Solutions Architect. This service is based on the AWS Well-Architected Framework, which was developed to help cloud architects build secure, high-performing, resilient, and efficient application infrastructure. You can review the state of your workloads and compare them to the latest AWS architectural best practices.

/

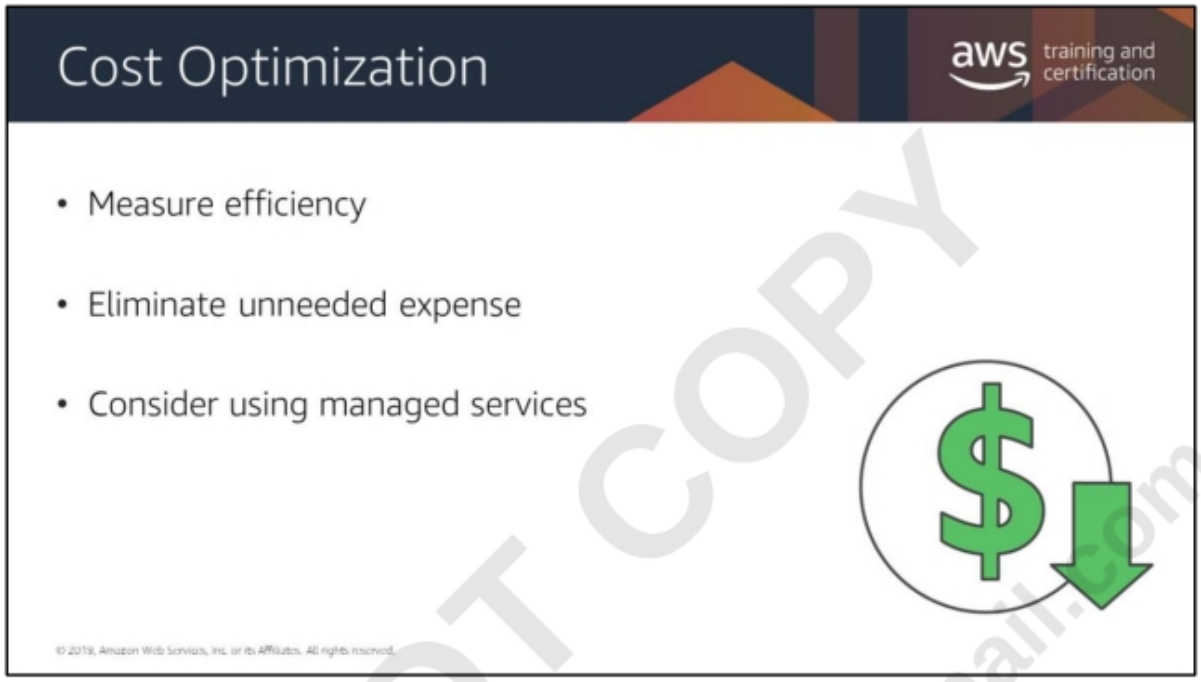
Security deals with protecting information and mitigating possible damage. Your architecture will present a much stronger security presence by implementing some basic security measures, like implementing a strong identity foundation, enabling traceability, applying security at all layers, automating security best practices, protecting data in transit and at rest

For more information, see https://d1.awsstatic.com/whitepapers/architecture/AWS-Security-Pillar.pdf



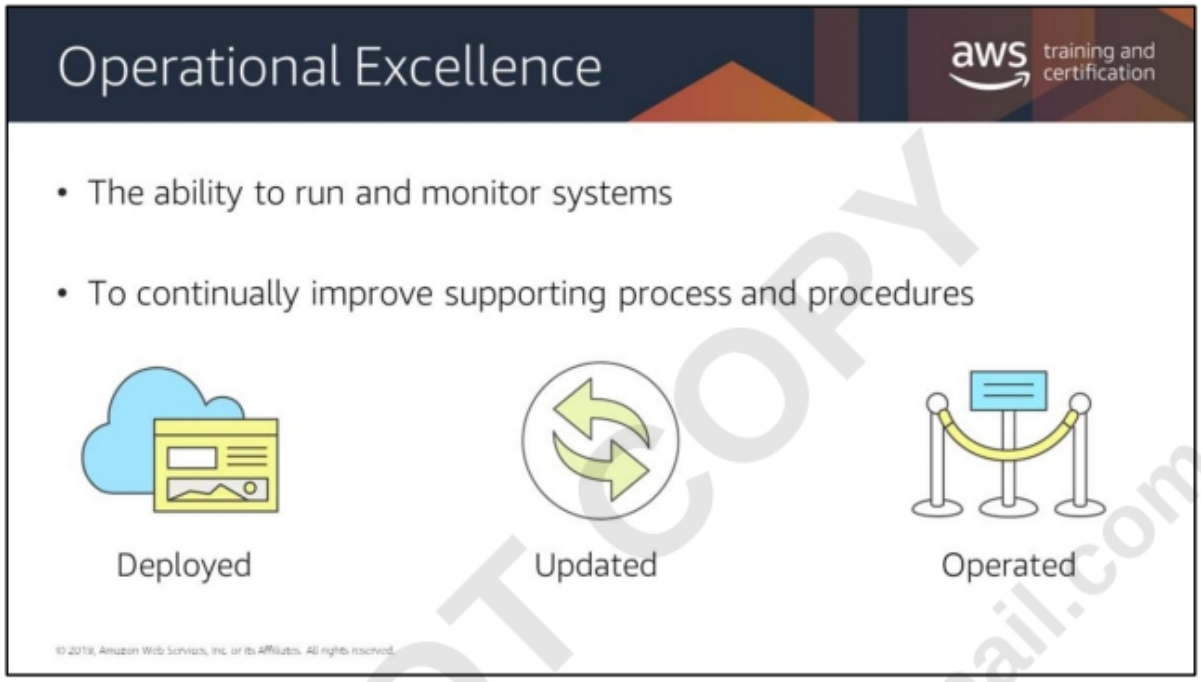
Ensuring reliability can be difficult in a traditional environment. Issues arise from single points of failure, lack of automation, and lack of elasticity. When applying the ideas of the reliability pillar, you will be able to prevent many of these issues. Properly designing your architecture in respect to high availability, fault tolerance, and overall redundancy will be helpful for you and your customers.

For more information, see https://d1.awsstatic.com/whitepapers/architecture/AWS-Reliability-Pillar.pdf



Cost optimization is an ongoing requirement of any good architectural design. The process is iterative and should be refined and improved throughout your production lifetime. Understanding how efficient your current architecture is in relation to your goals will ultimately help with removing unneeded expense. Consider using managed services as they operate at cloud scale and can offer a lower cost per transaction or service.

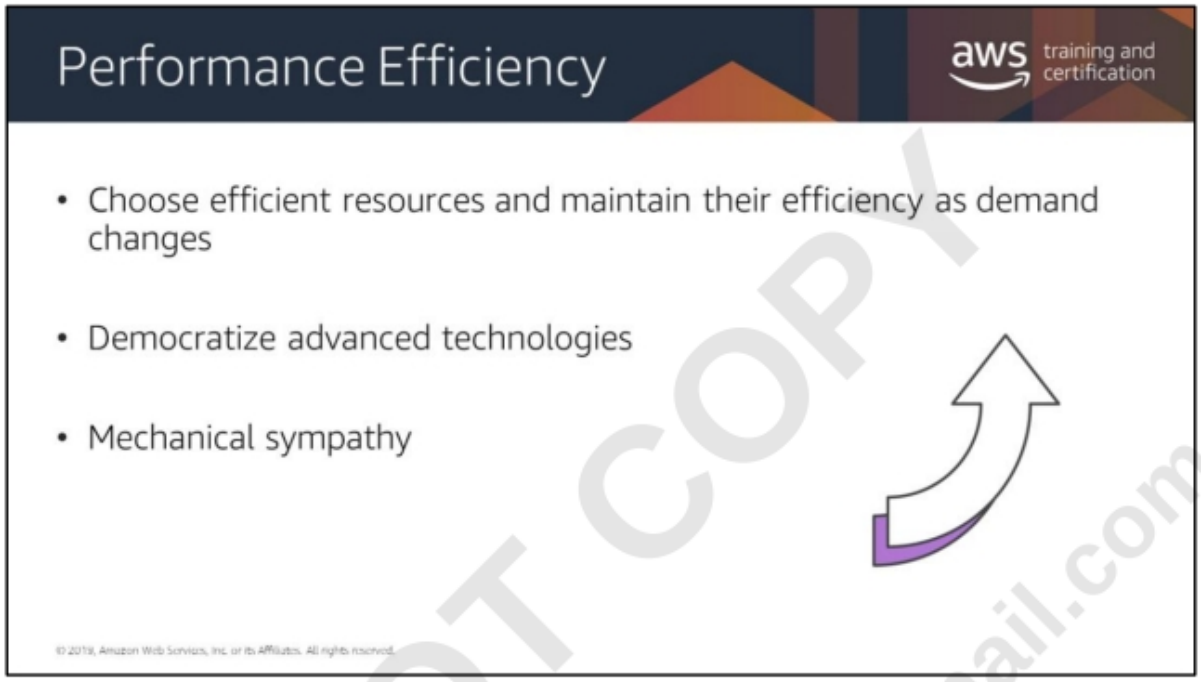
For more information, see https://d1.awsstatic.com/whitepapers/architecture/AWS-Cost-Optimization-Pillar.pdf



When creating a design or architecture, you must be aware of how it will be deployed, updated, and operated. It is imperative that you work towards defect reductions and safe fixes and enable observation with logging instrumentation.

In AWS, you can view your entire workload (applications, infrastructure, policy, governance, and operations) as code. It can all be defined in and updated using code. This means you can apply the same engineering discipline that you use for application code to every element of your stack.

For more information, see https://d1.awsstatic.com/whitepapers/architecture/AWS-Operational-Excellence-Pillar.pdf



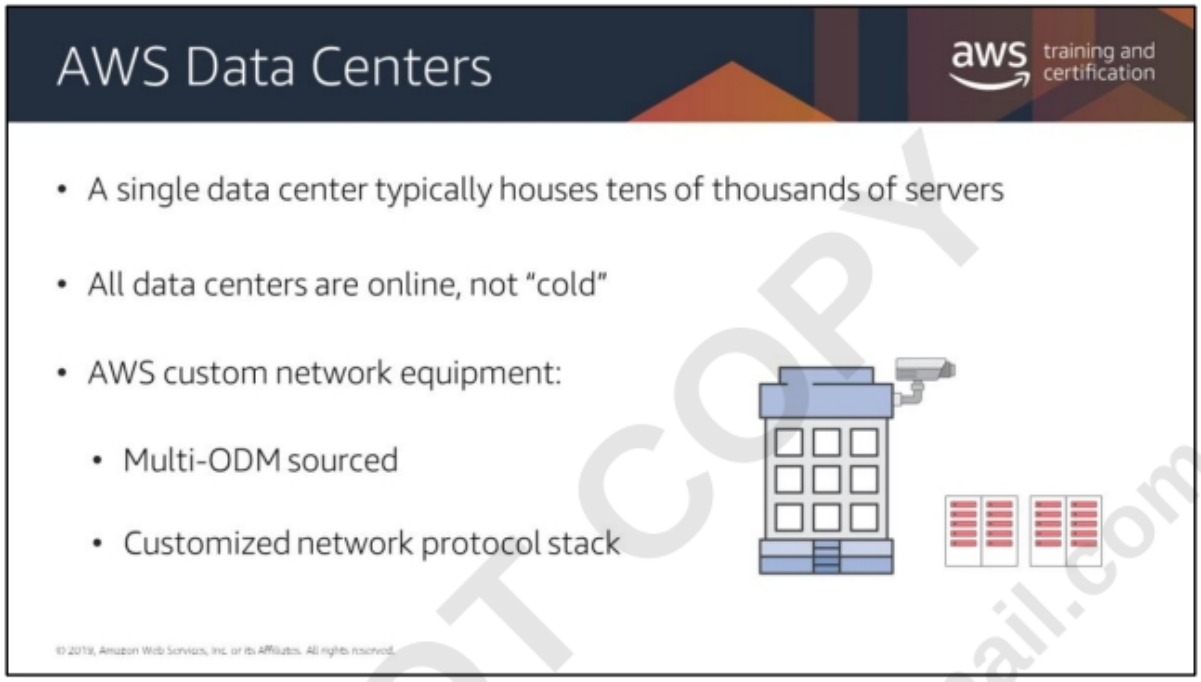
When considering performance, you want to maximize your performance by using computation resources efficiently and maintain that efficiency as the demand changes.

It is also important to democratize advanced technologies. In situations where technology is difficult to implement yourself, consider using a vendor. In implementing the technology for you, the vendor takes on the complexity and knowledge, freeing your team to focus on more value-added work.

Mechanical sympathy: Use the technology approach that aligns best to what you are trying to achieve. For example, consider data access patterns when you select database or storage approaches.

For more information, see https://d1.awsstatic.com/whitepapers/architecture/AWS-Performance-Efficiency-Pillar.pdf

/

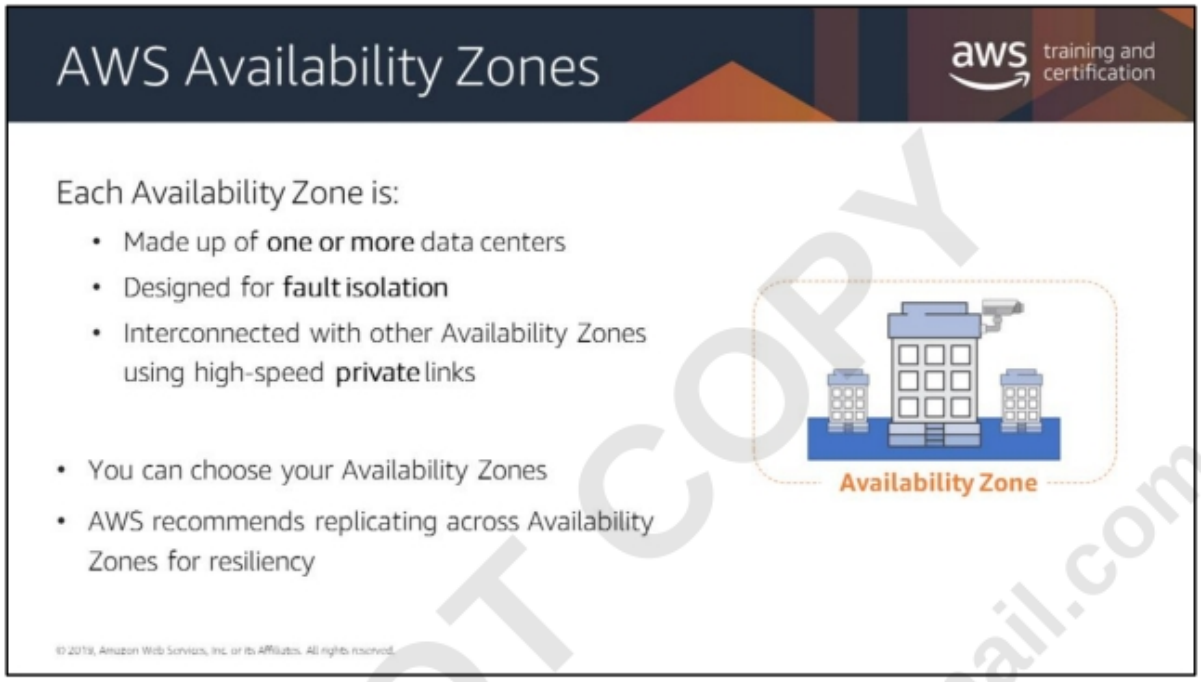


AWS data centers are built in clusters in various global regions. Larger data centers are undesirable; all data centers are online and serving customers. No data center is "cold;" in case of failure, automated processes move customer data traffic away from the affected area. Core applications are deployed in an N+1 configuration, so that in the event of a data center failure, there is sufficient capacity to enable traffic to be load-balanced to the remaining sites.

Original design manufacturers, or ODMs, designs and manufactures products based on specifications from a second company. The second company then rebrands the products for sale.

For more information, see https://aws.amazon.com/compliance/data-center/

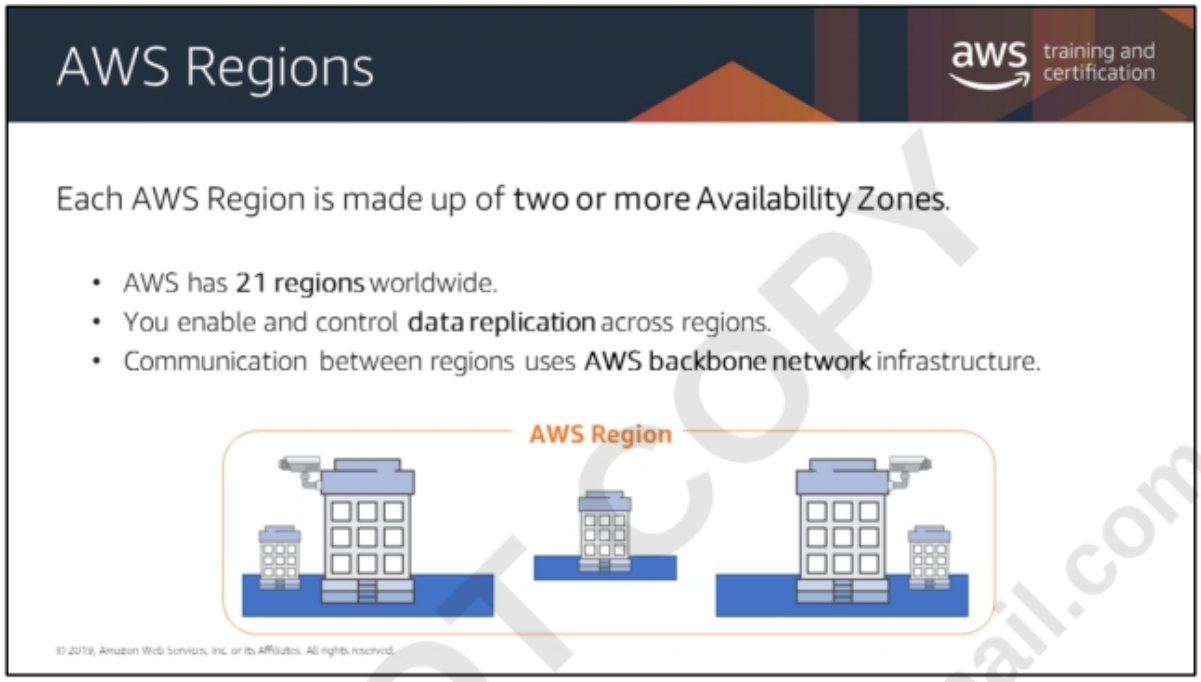
/



AWS data centers are organized into *Availability Zones*. Each Availability Zone comprises one or more data centers, with some Availability Zones having as many as six data centers. However, no data center can be part of two Availability Zones.

Each Availability Zone is designed as an independent failure zone. This means that Availability Zones are physically separated within a typical metropolitan region and are located in lower-risk flood plains (specific flood-zone categorization varies by region). In addition to having discrete uninterruptable power supply and onsite backup generation facilities, they are each fed via different grids from independent utilities to further reduce single points of failure. Availability Zones are all redundantly connected to multiple tier-1 transit providers.

You are responsible for selecting the Availability Zones where your systems will reside. Systems can span multiple Availability Zones. You should design your systems to survive temporary or prolonged failure of an Availability Zone if a disaster occurs. Distributing applications across multiple Availability Zones allows them to remain resilient in most failure situations, including natural disasters or system failures.

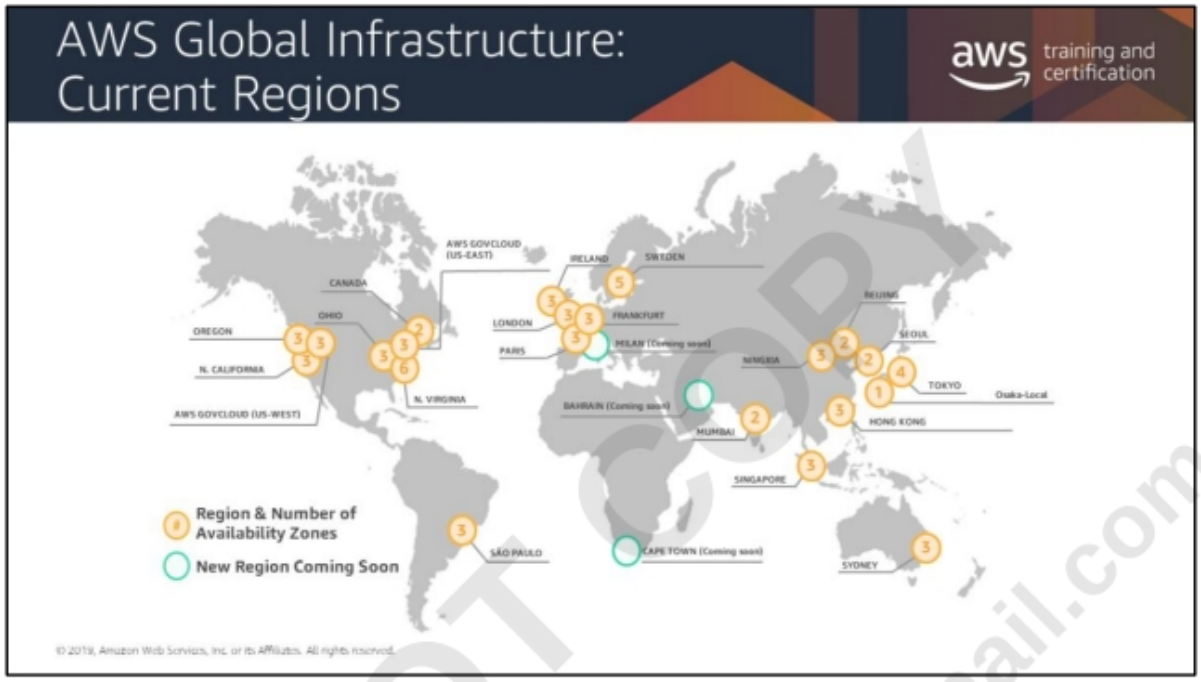


Availability Zones are further grouped into *AWS Regions*. Each region contains two or more Availability Zones.

When you distribute applications across multiple Availability Zones, be aware of location-dependent privacy and compliance requirements, such as the EU Data Privacy Directive. When you store data in a specific region, it is not replicated outside that region. AWS never moves your data out of the region you put it in. It is your responsibility to replicate data across regions, if your business needs require that. AWS provides information about the country, and—where applicable—the state where each region resides; you are responsible for selecting the region to store data in based on your compliance and network latency requirements.

AWS Regions are connected to multiple Internet Service Providers (ISPs) as well as to a private global network backbone, which provides lower cost and more consistent cross-region network latency when compared with the public Internet.

For more information, see https://aws.amazon.com/about-aws/global-infrastructure/#reglink-pr



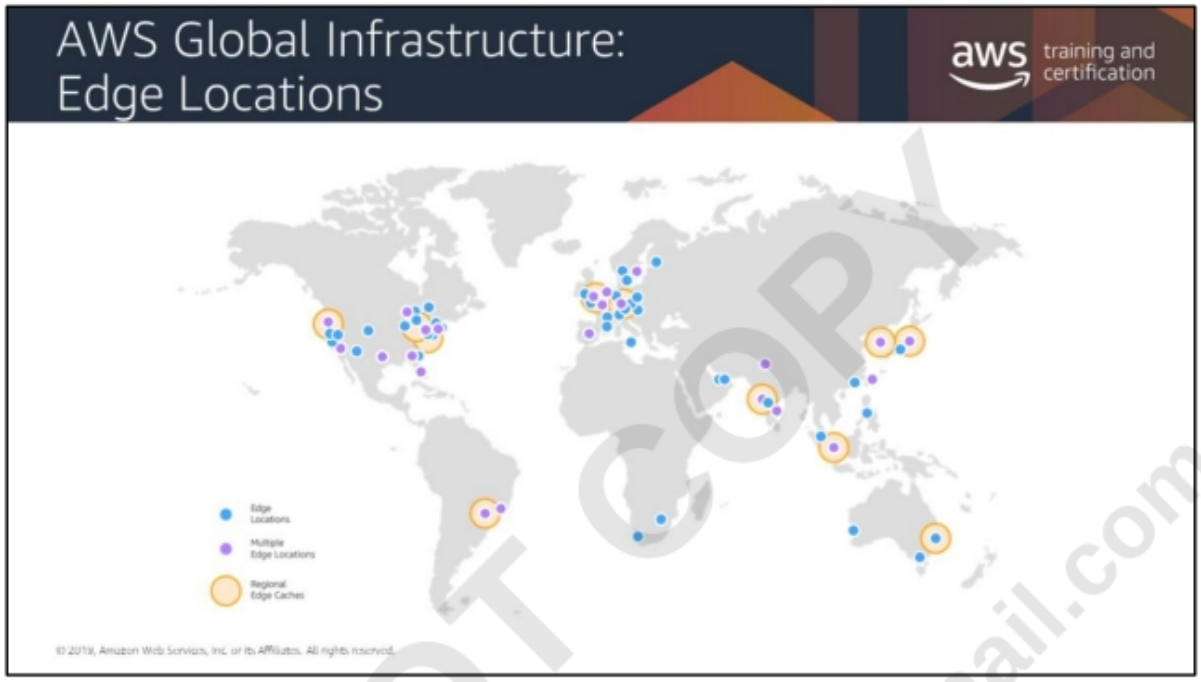
AWS is steadily expanding its global infrastructure to help customers who want to ensure that their data resides only in the region they specify and to achieve lower latency and higher throughput. As you and all customers grow your businesses, AWS will continue to provide infrastructure that meets your global requirements.

The isolated GovCloud (US) Region is designed to allow US government agencies and customers to move sensitive workloads into the cloud by addressing their specific regulatory and compliance requirements.

AWS products and services are available by region so you may not see all regions available for a given service.

You can run applications and workloads from a region to reduce latency to end users while avoiding the up-front expenses, long-term commitments, and scaling challenges associated with maintaining and operating a global infrastructure.

/



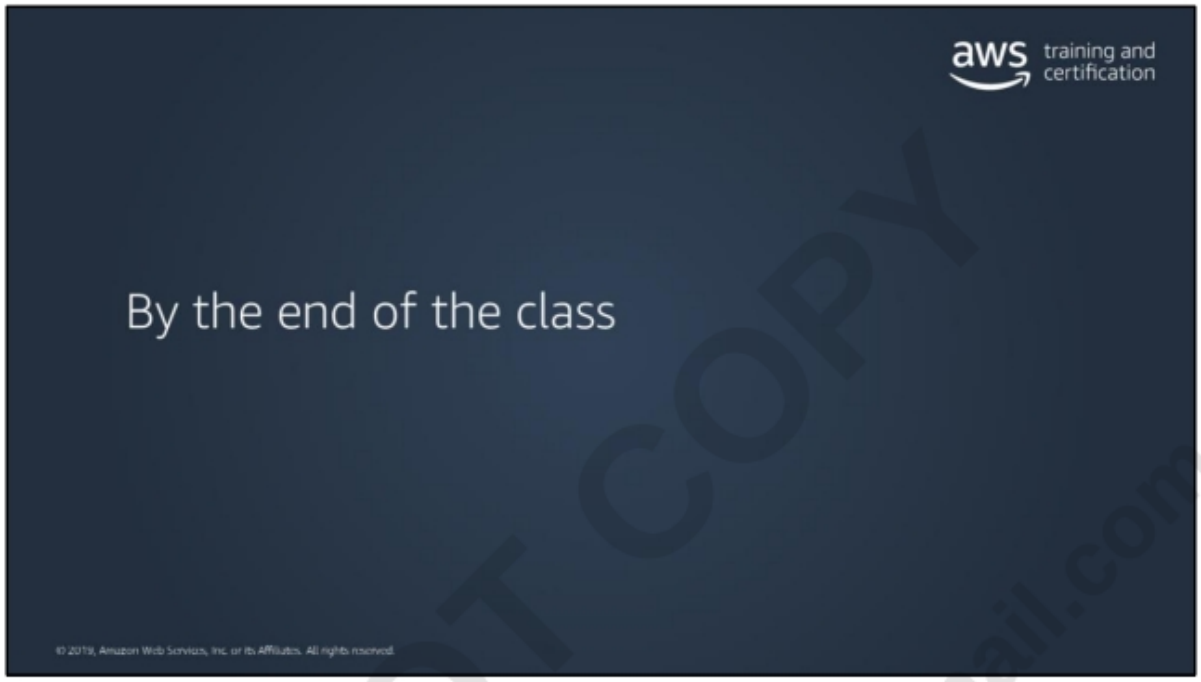
To deliver content to end users with lower latency, Amazon CloudFront uses a global network of 187 Points of Presence (176 Edge Locations and 11 Regional Edge Caches) in 69 cities across 30 countries.

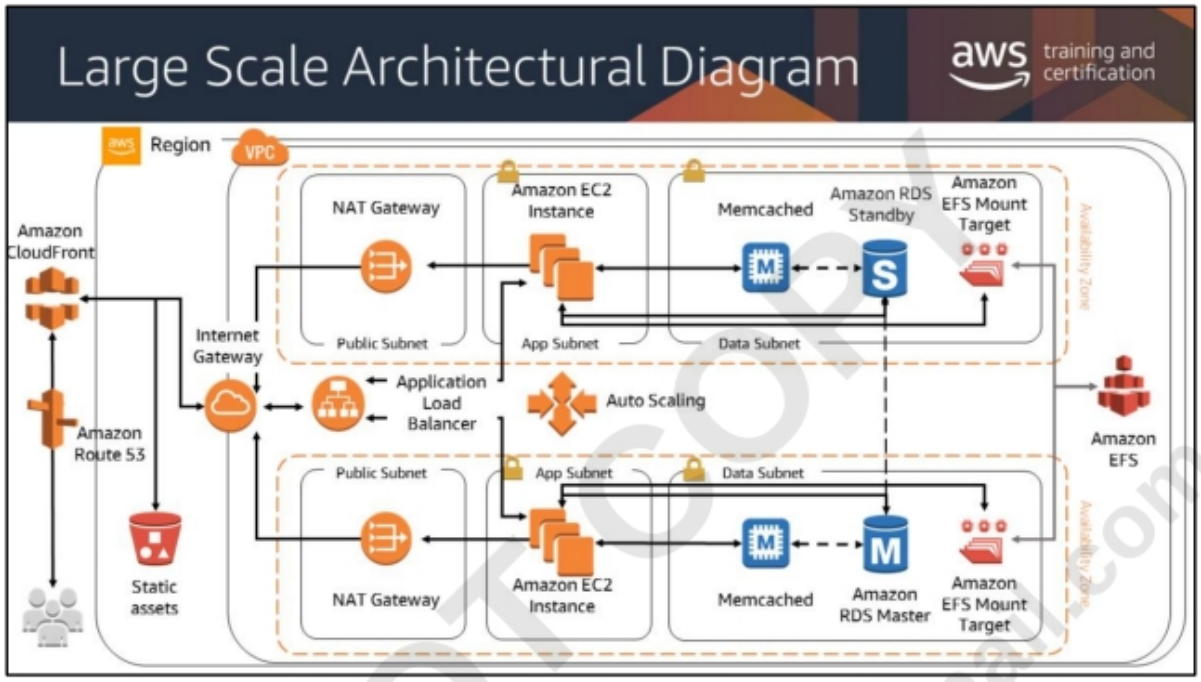
Edge locations are located in: North America, Europe, Asia, Australia, and South America, and support AWS services like Amazon Route 53 and Amazon CloudFront.

Regional Edge Caches

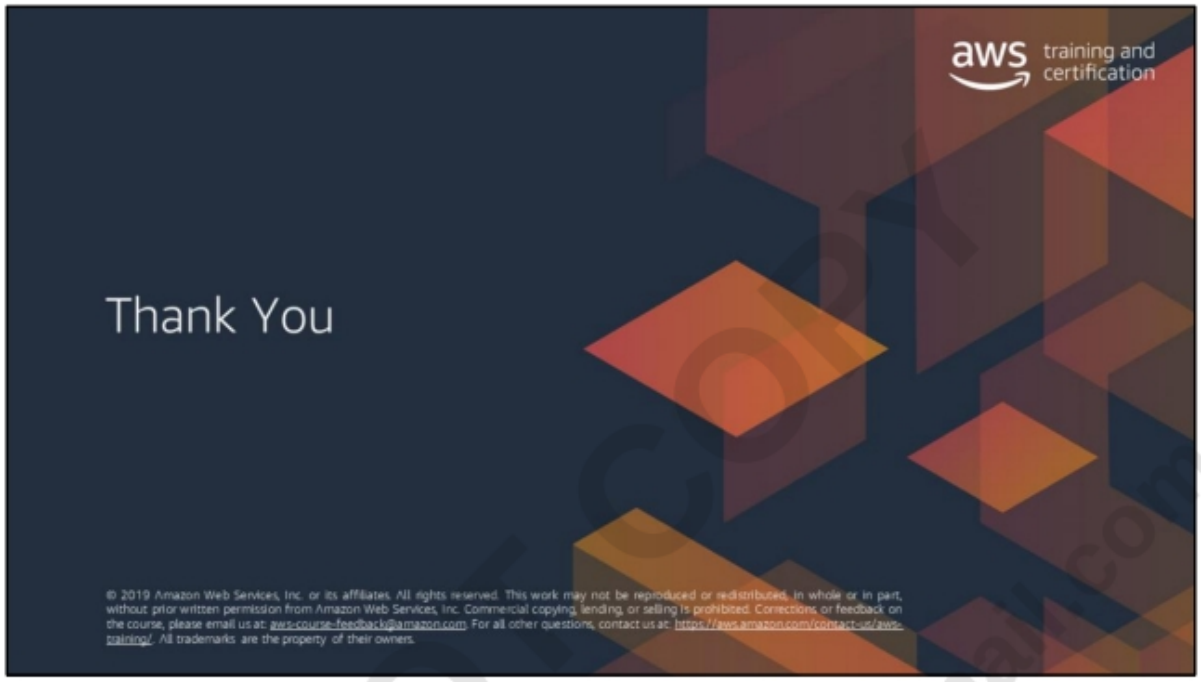
Regional edge caches, used by default with Amazon CloudFront, are utilized when you have content that is not accessed frequently enough to remain in an edge location. Regional edge caches absorb this content and provide an alternative to that content having to be fetched from the origin server.

For more information, see https://aws.amazon.com/cloudfront/features/



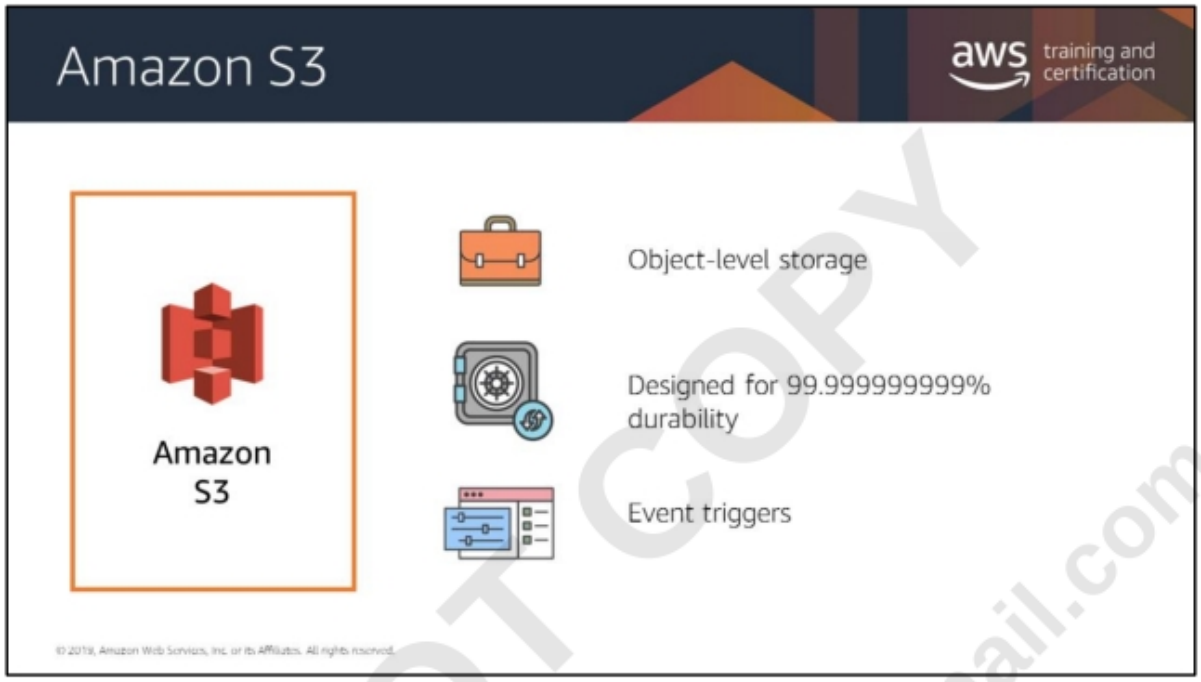


By the end of class, you will be able to understand all of the components of this architectural diagram. You will also be able to construct your own architectural solutions that are just as large and robust.



/

By the end of class, you will be able to understand all of the components of this architectural diagram. You will also be able to construct your own architectural solutions that are just as large and robust.



Amazon S3 is *object-level storage*, which means that if you want to change a part of a file, you have to make the change and then re-upload the entire modified file.

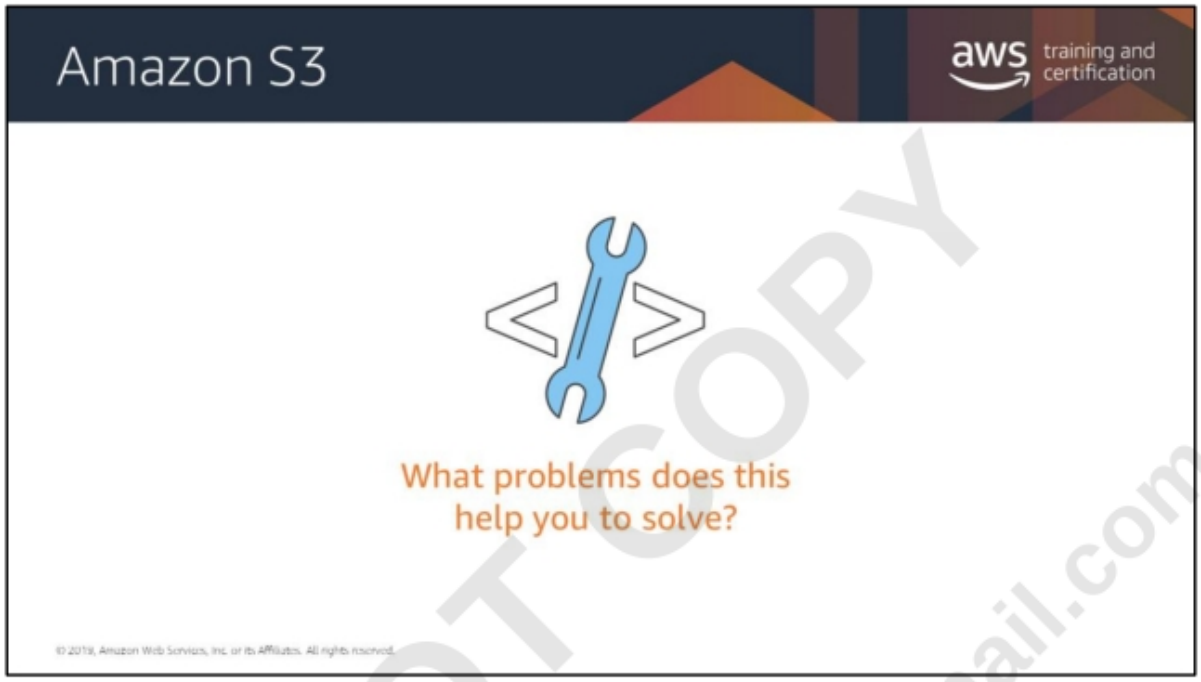
Amazon S3 allows you to store as much data as you want. Individual objects cannot be larger than 5 TB; however, you can store as much total data as you need.

By default, data in Amazon S3 is stored redundantly across multiple facilities and multiple devices in each facility.

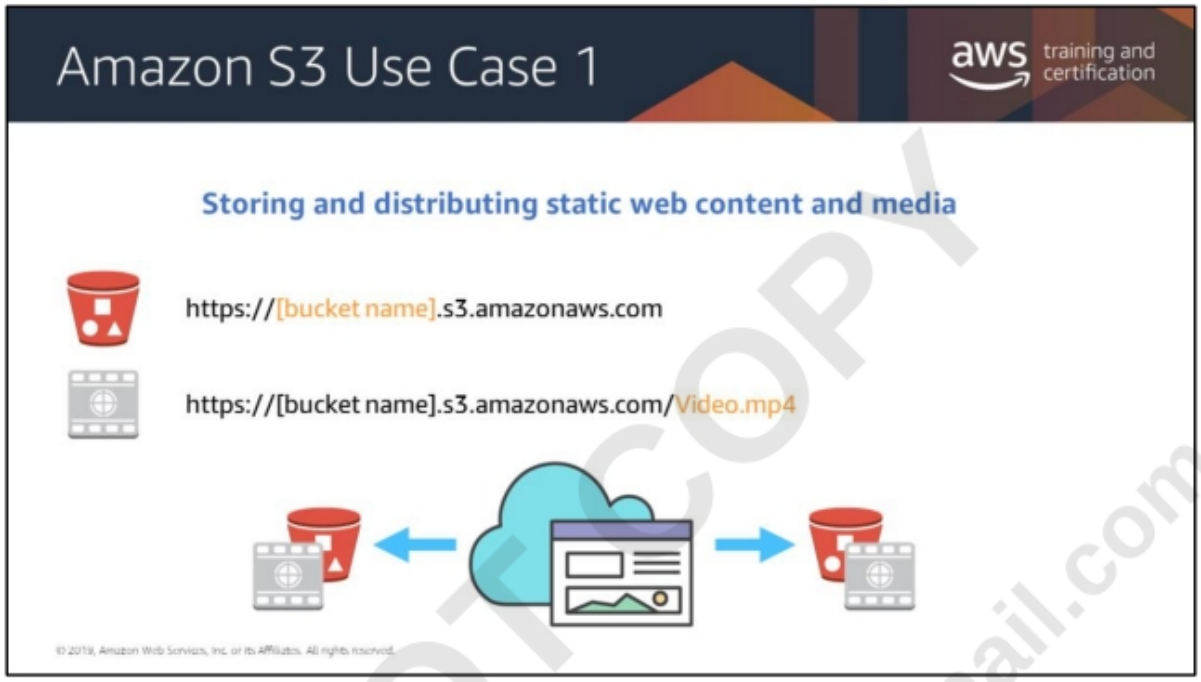
Amazon S3 can be accessed via the web-based AWS Management Console, programmatically via the API and SDKs, or with third-party solutions (which use the API/SDKs).

Amazon S3 includes *event notifications* that allow you to set up automatic notifications when certain events occur, such as an object being uploaded to or deleted from a specific bucket. Those notifications can be sent to you, or they can be used to trigger other processes, such as AWS Lambda scripts.

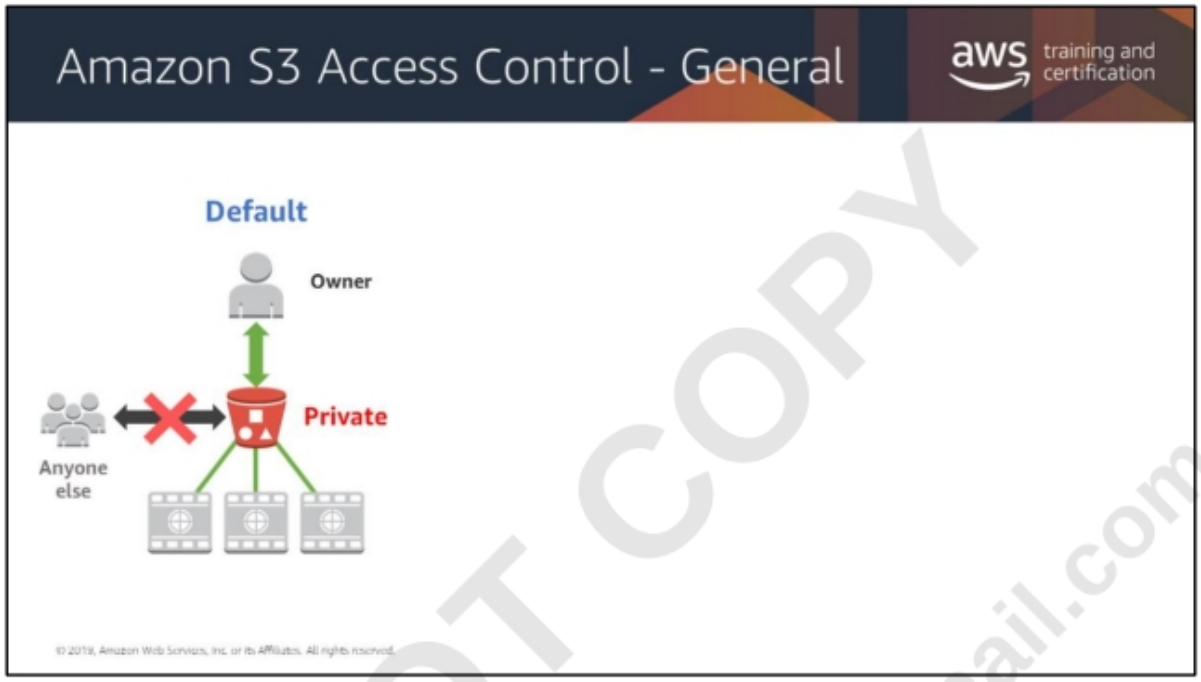
With *storage class analysis*, you can analyze storage access patterns and transition the right data to the right storage class. This new S3 Analytics feature automatically identifies the optimal lifecycle policy to transition less frequently accessed storage to Amazon S3 Standard-Infrequent Access (S3 Standard-IA). You can configure a storage class analysis policy to monitor an entire bucket, a prefix, or object tag. Once an infrequent access pattern is observed, you can easily create a new lifecycle age policy based on the results. Storage class analysis also provides daily visualizations of your storage usage in the AWS Management Console. You can export these to an S3 bucket to analyze using the business intelligence tools of your choice, such as Amazon QuickSight.



So how can you use these features of Amazon S3 to address your needs?



First, you can use Amazon S3 to store and distribute static web content or media. These files can be delivered directly from Amazon S3 because each object is associated with a unique HTTP URL, which must be DNS compliant (e.g. JohnJSmith.com). Amazon S3 can also be used as an origin for a content delivery network (such as Amazon CloudFront). Amazon S3 works well for fast-growing websites that require strong elasticity. This might include workloads with large amounts of user generated content, such as video or photo sharing.



By default, all Amazon S3 resources—buckets, objects, and related sub-resources (for example, lifecycle configuration and website configuration)—are private: only the resource owner, an AWS account that created it, can access the resource. The resource owner can grant access permissions to others by writing an access policy.

Module 7 covers AWS Identity and Access Management (IAM) vs. access control lists (ACL) and bucket policies. For more information about access control in Amazon S3, see https://docs.aws.amazon.com/AmazonS3/latest/dev/access-control-overview.html

IMPORTANT!
While the static website use case with S3 for static content is a great example of quickly getting set up with an AWS architecture, that public access to Amazon S3 is not the majority of use cases. ***Most use cases DO NOT require public access.*** *More often, Amazon S3 stores data that is part of another application. Public access should never be used for these types of buckets.*

Amazon S3 buckets are **protected by default**. The only entities with access to a newly created, unmodified bucket are the account administrator and root user.

Modifications to bucket policies can enable additional access, and AWS provides a number of different tools to enable developers to configure buckets for a wide variety of workloads. Amazon S3 includes a "block public access" feature, which acts as an additional layer of protection to prevent accidental exposure of customer data.
In the public access settings for a bucket, customers can specify the following four options. All options are enabled by default.

- Block new public ACLs and uploading public objects.
- Remove public access granted though public ACLs.
- Block new public bucket policies.
- Block public and cross-account access to buckets that have public policies.

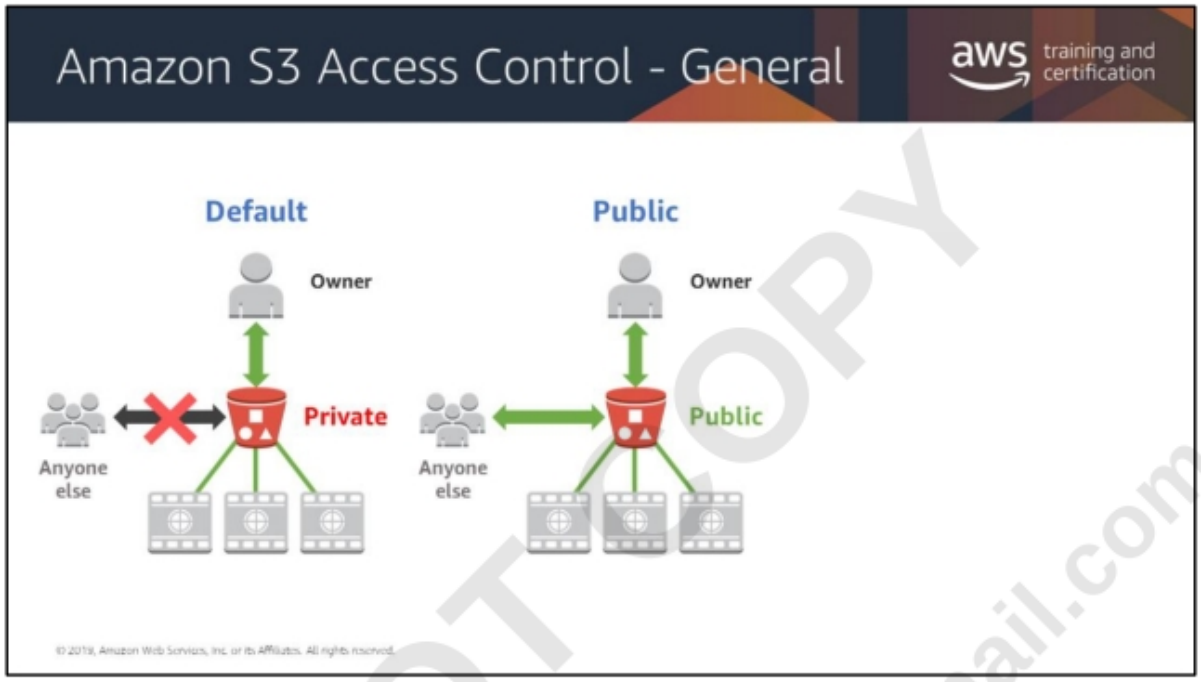
Public Access Settings
These settings must be manually disabled for public access settings, such as with a public, static website.
https://aws.amazon.com/blogs/aws/amazon-s3-block-public-access-another-layer-of-protection-for-your-accounts-and-buckets/

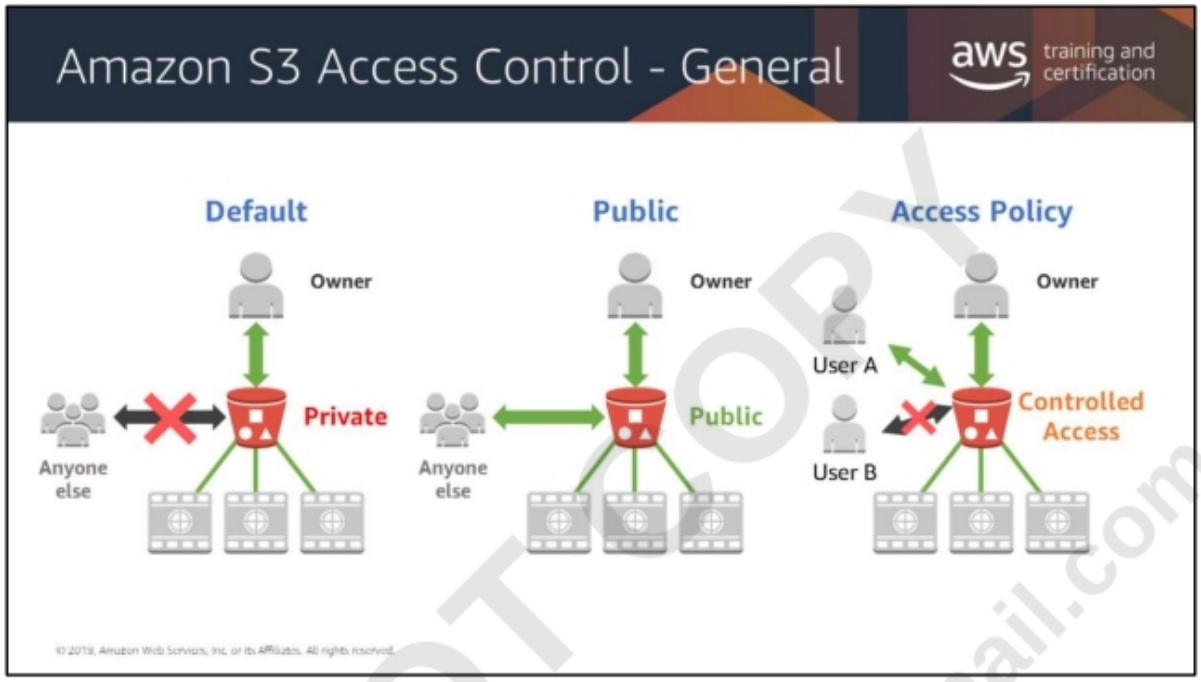
To understand more about Public and Private access settings, see:
https://youtu.be/x25FSsXrBqU?t=989 (start at 16:29)

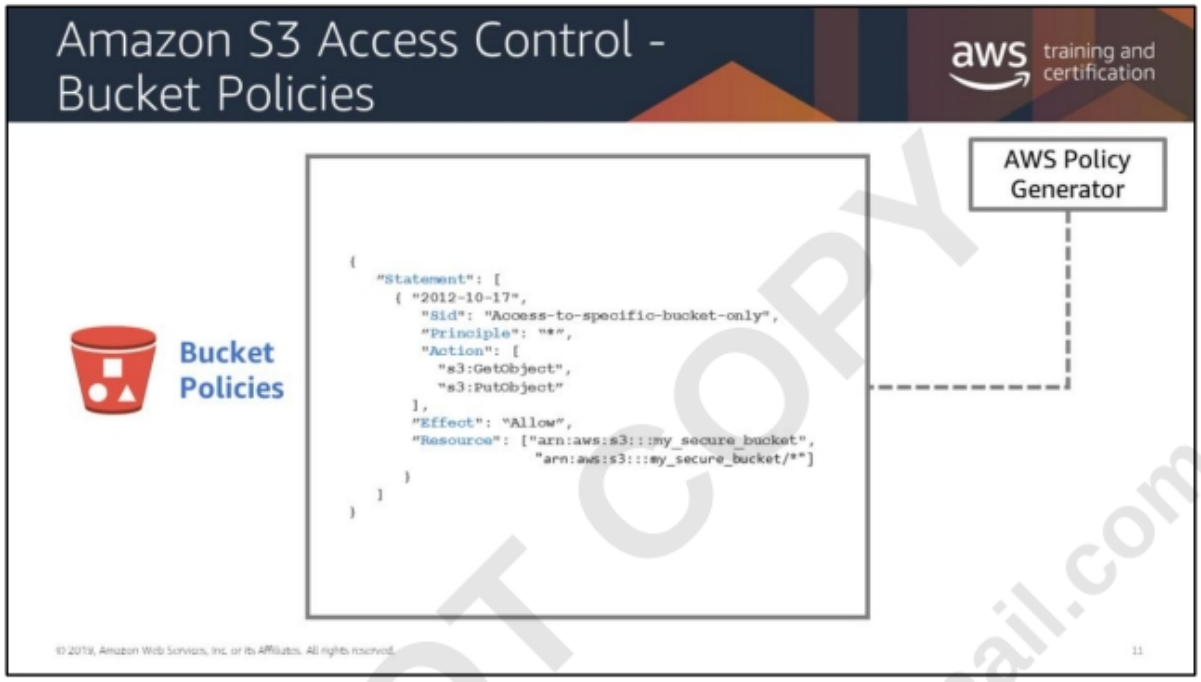
Please also read:
Jeff Barr's blog post from November 2018
https://aws.amazon.com/blogs/aws/amazon-s3-block-public-access-another-layer-of-protection-for-your-accounts-and-buckets/

S3 Developer Guide: Using Amazon S3 Block Public Access
https://docs.aws.amazon.com/AmazonS3/latest/dev/access-control-block-public-access.html

S3 Console User Guide: How Do I Block Public Access to S3 Buckets?
https://docs.aws.amazon.com/AmazonS3/latest/user-guide/block-public-access.html

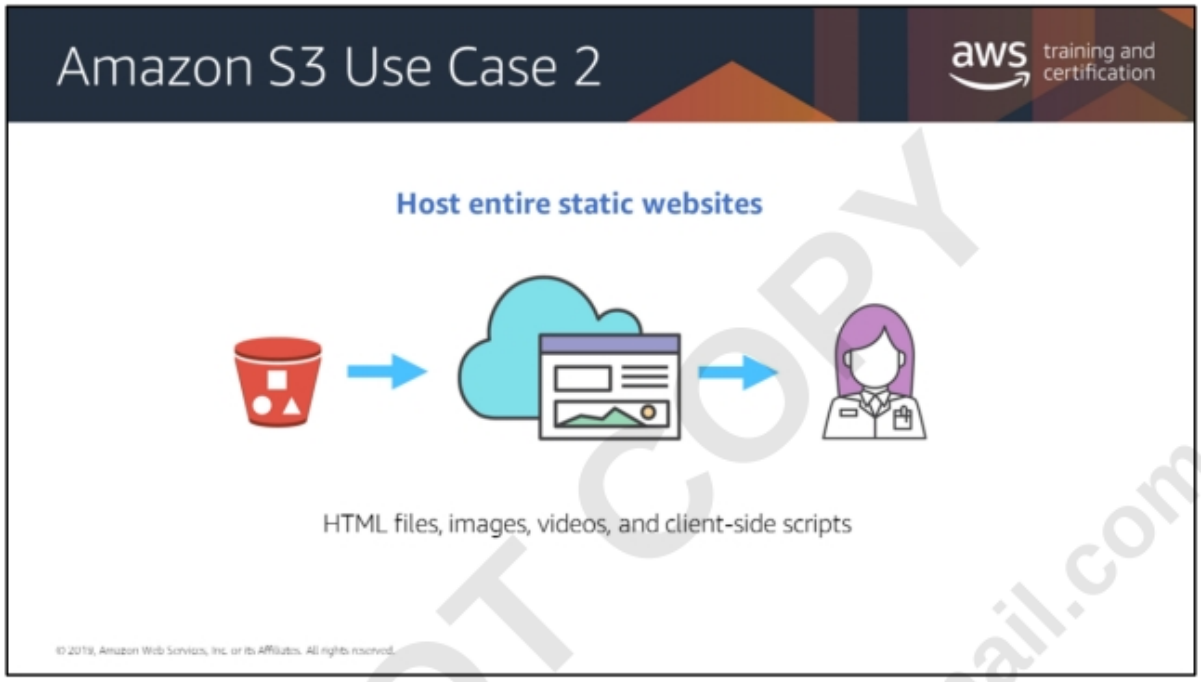




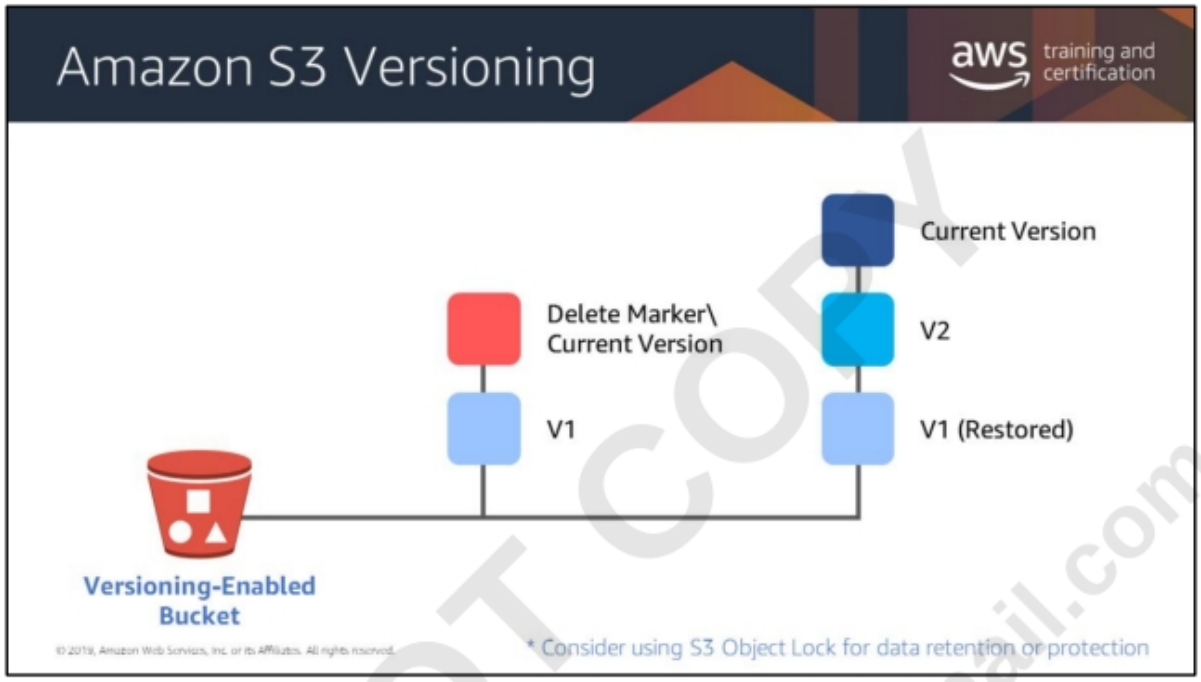


In your S3 buckets, you can add polices to allow other AWS accounts or users to access the objects stored within. Bucket policies can supplement and, in some cases, replace standard ACL access policies.

Bucket policies are limited to 20 KB in size.



You can use Amazon S3 to host entire static websites. Amazon S3 provides a low-cost, highly available, and highly scalable solution, including storage for static HTML files, images, videos, and client-side scripts in formats such as JavaScript.



Versioning-enabled buckets enable you to recover objects from accidental deletion or overwrite. For example:

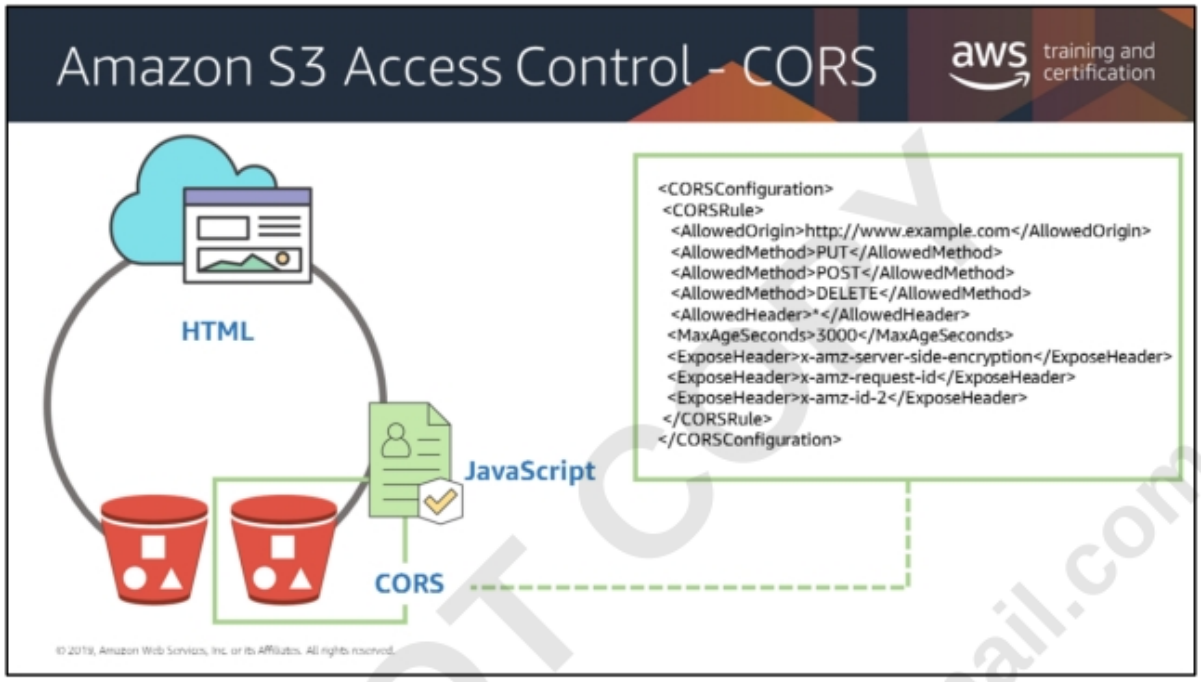
- If you delete an object, instead of removing it permanently, Amazon S3 inserts a *delete marker*, which becomes the current object version. You can always restore the previous version.
- If you overwrite an object, it results in a new object version in the bucket. You can always restore the previous version.

For more information about versioning, see
https://docs.aws.amazon.com/AmazonS3/latest/dev/Versioning.html

You can use S3 Object Lock for data retention or protection. By using the Write-Once-Read-Many (WORM) model, you can prevent accidental overwrites or deletions within S3 storage.

Use Retention Periods for locking an object for a fixed period of time, or a Legal Hold for a lock until explicitly removed.

This feature works only on versioned buckets with the retention periods and legal holds applying to individual object versions and Amazon S3 stores the lock information in the metadata for that object version. This does not prevent a new version from being created. Object Lock helps comply with **SEC 17a-4, CTCC, and FINRA.**
https://docs.aws.amazon.com/AmazonS3/latest/dev/object-lock.html



Cross-origin resource sharing (CORS) defines a way for client web applications that are loaded in one domain to interact with resources in a different domain. With CORS support, you can build rich client-side web applications with Amazon S3 and selectively allow cross-origin access to your Amazon S3 resources.

To configure your bucket to allow cross-origin requests, you create a CORS configuration, which is an XML document with rules that identify:

- The origins that you will allow to access your bucket.
- The operations (HTTP methods) that will support for each origin.
- Other operation-specific information.

For more information about CORS, see
https://docs.aws.amazon.com/AmazonS3/latest/dev/cors.html