

Introducción a



AWS - ¿Qué es?

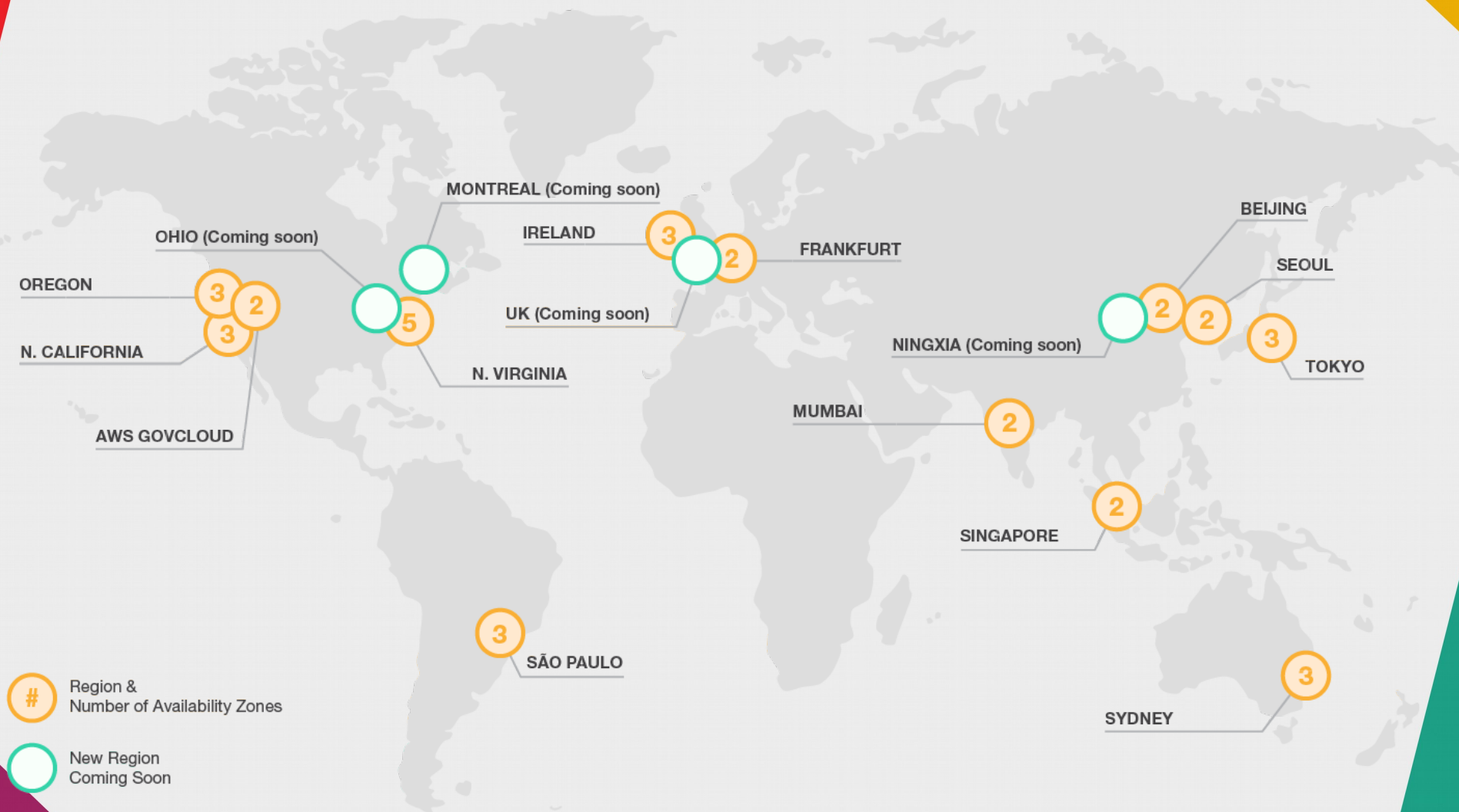
- Spin-off del área de infraestructura de Amazon.com (2006)
- Definieron el concepto “Cloud Computing”
- Líder del mercado [1][2]
- IaaS y PaaS principalmente, SaaS también
- Servicios y Costos “Elásticos”
- Amazon, Netflix, Spotify, Instagram, LinkedIn*, Dropbox*, Airbnb, Reddit... Frávega, Etermax...

[1] <http://www.businessinsider.com/why-amazon-is-so-hard-to-topple-in-the-cloud-and-where-everybody-else-falls-2015-10>

[2] <https://www.srgresearch.com/articles/aws-remains-dominant-despite-microsoft-and-google-growth-surges>





AWS - ¿Qué es?

13 Regiones, 35 Zonas de Disponibilidad, 59 Edge Locations







AWS - Servicios [1]

Broad & Deep Core Cloud Infrastructure Services

 Compute	 Storage & Content Delivery	 Database	 Networking
Virtual Servers	Object Storage	Relational	Virtual Private Cloud
Containers	CDN	Database Migration	Direct Connections
1-Click Web App Deployment	Block Storage	NoSQL	Load Balancing
Event-Driven Compute Functions	File System Storage	Caching	DNS
Auto Scaling	Archive Storage	Data Warehouse	
Load Balancing	Data Transport		
	Integrated Storage		












Accelerate your Cloud Success with Rich Platform Services

 Analytics	 Enterprise Applications	 Mobile Services	 Internet of Things
Business Intelligence	Desktop Virtualization	Mobile Development	IoT Platform
Data Warehouse	Email & Calendaring	API Management	Device SDK
Machine Learning	Document Sharing & Feedback	Identity	Registry
Streaming Data		App Testing	Device Shadows
Elasticsearch		Mobile Analytics	Rules Engine
Hadoop		Development	
Data Pipelines		Notifications	

Increase Developer Productivity and Operational Efficiency





 Developer Tools	 Management Tools	 Security & Identity	 Application Services
Source Code Management	Monitoring & Logs	Access Control	API Management
Code Deployment	Resource Templates	SSL/TLS Certificates	App Streaming
Continuous Delivery	Usage & Resource Auditing	Key Storage & Management	Search
	Dev/Ops Resource Management	Identity Management	Transcoding
	Service Catalog	Security Assessment	Email
	Performance Optimization	Web Application Firewall	Notifications
			Queueing
			Workflow

AWS - Servicios







 **AWS**  **Services**   **EC2**  **Route 53**  **S3**  **CloudFront**  **API Gateway**  **IAM**  **CloudTrail** **Edit** 

Amazon Web Services






Compute

-  **EC2**
Virtual Servers in the Cloud
-  **EC2 Container Service**
Run and Manage Docker Containers
-  **Elastic Beanstalk**
Run and Manage Web Apps
-  **Lambda**
Run Code in Response to Events




Storage & Content Delivery

-  **S3**
Scalable Storage in the Cloud
-  **CloudFront**
Global Content Delivery Network
-  **Elastic File System**
Fully Managed File System for EC2
-  **Glacier**
Archive Storage in the Cloud
-  **Snowball**
Large Scale Data Transport
-  **Storage Gateway**
Hybrid Storage Integration




Database

-  **RDS**
Managed Relational Database Service
-  **DynamoDB**
Managed NoSQL Database
-  **ElastiCache**
In-Memory Cache
-  **Redshift**
Fast, Simple, Cost-Effective Data Warehousing
-  **DMS**
Managed Database Migration Service








Networking

-  **VPC**
Isolated Cloud Resources
-  **Direct Connect**
Dedicated Network Connection to AWS
-  **Route 53**
Scalable DNS and Domain Name Registration






Developer Tools

-  **CodeCommit**
Store Code in Private Git Repositories
-  **CodeDeploy**
Automate Code Deployments
-  **CodePipeline**
Release Software using Continuous Delivery






Management Tools

-  **CloudWatch**
Monitor Resources and Applications
-  **CloudFormation**
Create and Manage Resources with Templates
-  **CloudTrail**
Track User Activity and API Usage
-  **Config**
Track Resource Inventory and Changes
-  **OpsWorks**
Automate Operations with Chef
-  **Service Catalog**
Create and Use Standardized Products
-  **Trusted Advisor**
Optimize Performance and Security


Security & Identity

-  **Identity & Access Management**
Manage User Access and Encryption Keys
-  **Directory Service**
Host and Manage Active Directory
-  **Inspector**
Analyze Application Security
-  **WAF**
Filter Malicious Web Traffic
-  **Certificate Manager**
Provision, Manage, and Deploy SSL/TLS Certificates


Analytics

-  **EMR**
Managed Hadoop Framework
-  **Data Pipeline**
Orchestration for Data-Driven Workflows
-  **Elasticsearch Service**
Run and Scale Elasticsearch Clusters
-  **Kinesis**
Work with Real-Time Streaming Data
-  **Machine Learning**
Build Smart Applications Quickly and Easily






Internet of Things

-  **AWS IoT**
Connect Devices to the Cloud








Game Development

-  **GameLift**
Deploy and Scale Session-based Multiplayer Games




Mobile Services

-  **Mobile Hub**
Build, Test, and Monitor Mobile Apps
-  **Cognito**
User Identity and App Data Synchronization
-  **Device Farm**
Test Android, iOS, and Web Apps on Real Devices in the Cloud
-  **Mobile Analytics**
Collect, View and Export App Analytics
-  **SNS**
Push Notification Service

Application Services

-  **API Gateway**
Build, Deploy and Manage APIs
-  **AppStream**
Low Latency Application Streaming
-  **CloudSearch**
Managed Search Service
-  **Elastic Transcoder**
Easy-to-Use Scalable Media Transcoding
-  **SES**
Email Sending and Receiving Service
-  **SQS**
Message Queue Service
-  **SWF**
Workflow Service for Coordinating Application Components

Enterprise Applications

-  **WorkSpaces**
Desktops in the Cloud
-  **WorkDocs**
Secure Enterprise Storage and Sharing Service
-  **WorkMail**
Secure Email and Calendaring Service

AWS EC2 - Tipos de Instancias



- **T2**: Burstable performance
máx. 2 vCPUs, 8 GB RAM, variable IOPS EBS
- **M3 / M4**: General purpose
máx. 40 vCPUs, 160 GB RAM, 4Gbps BW EBS
- **C3 / C4**: Compute optimized
máx. 36 vCPUs, 60 GB RAM, 4Gbps BW EBS
- **R3 / X1**: Memory optimized
máx. 128 vCPUs, 2 TB RAM, 10Gbps BW EBS
- **G2**: GPU compute
máx. 4 Nvidia CUDA GPUs/4GB RAM, 32 vCPUs, 60 GB RAM
- **D2 / I2**: Storage optimized
I2: High random I/O instances, máx 32 VCPUs, 244 GB RAM, 8x800 SSD
D2: Dense storage instances, máx 36 VCPUs, 244 GB RAM, 24x2TB HDD

<https://aws.amazon.com/ec2/instance-types/>

AWS EC2 - Modelo de Precios



- Capa Gratuita (Free Usage Tier)
- Instancias **On-Demand**
Pago por hora de ejecución.
- Instancias **Reservadas**
Pago por adelantado o en cuotas de un tipo de instancia.
- Instancias de Subasta (**Spot** Instances)
Se especifica el precio que se está dispuesto a pagar, y la/las instancias se asignan automáticamente una vez que el costo es menor a este valor.

<https://aws.amazon.com/ec2/pricing/>
<http://calculator.s3.amazonaws.com/index.html>

AWS EC2 - Servicios asociados

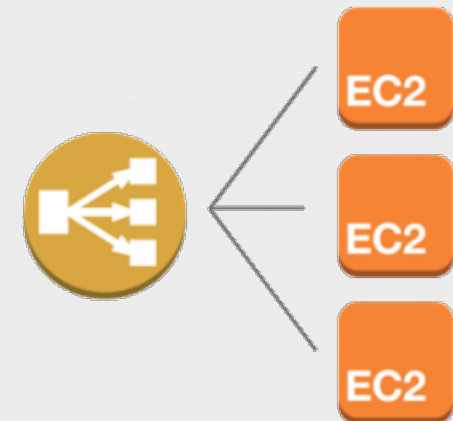
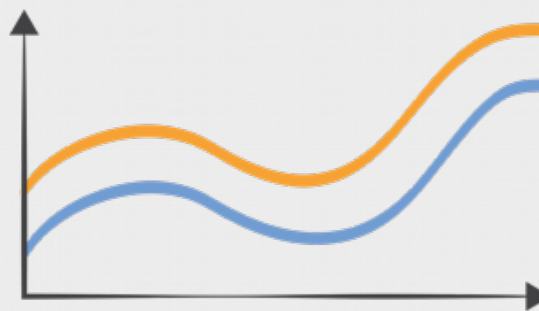
- **VPC** - Virtual Private Cloud

Direccionamiento IP privado/público estático/dinámico, subredes, gateways, ACLs, Security Groups, VPNs, VPC endpoint for S3, Direct Connect*...

- **Autoscaling**

- **ELB** - Elastic Load Balancing

HTTP/HTTPS/TCP, Application Load Balancer* (Layer-7 Routing, HTTP/2+Websockets support, Sticky Sessions...)



Almacenamiento y Entrega de Contenido

- **EBS** – Elastic Block Storage for EC2
- **S3** – Simple Storage Service
Ilimitados Objetos $\leq 5\text{TB}$, durabilidad y disponibilidad, replicación entre regiones, versionado, políticas de ciclo de vida y de acceso/seguridad, integración con el resto del ecosistema...
- **Glacier** – Archiving and Backup
- **CloudFront** – Content Delivery Network
Global, SSL offloading, diferentes backends, RTMP...
- **Route53** – Global DNS Service
Routing Policies, Health checks y failover, Geo DNS, integración con S3/Cloudfront, DNS privado para el VPC...



Bases de Datos

- **RDS** – Relational Database Service
Oracle, SQL Server, PostgreSQL, MySQL, MariaDB, AWS Aurora
- **DynamoDB** – NoSQL Database Service
- **ElastiCache** – In-Memory Caching Service
Redis, Memcache
- **Redshift** – Data Warehouse Service
BI analysis, Columnar Storage



DEMO

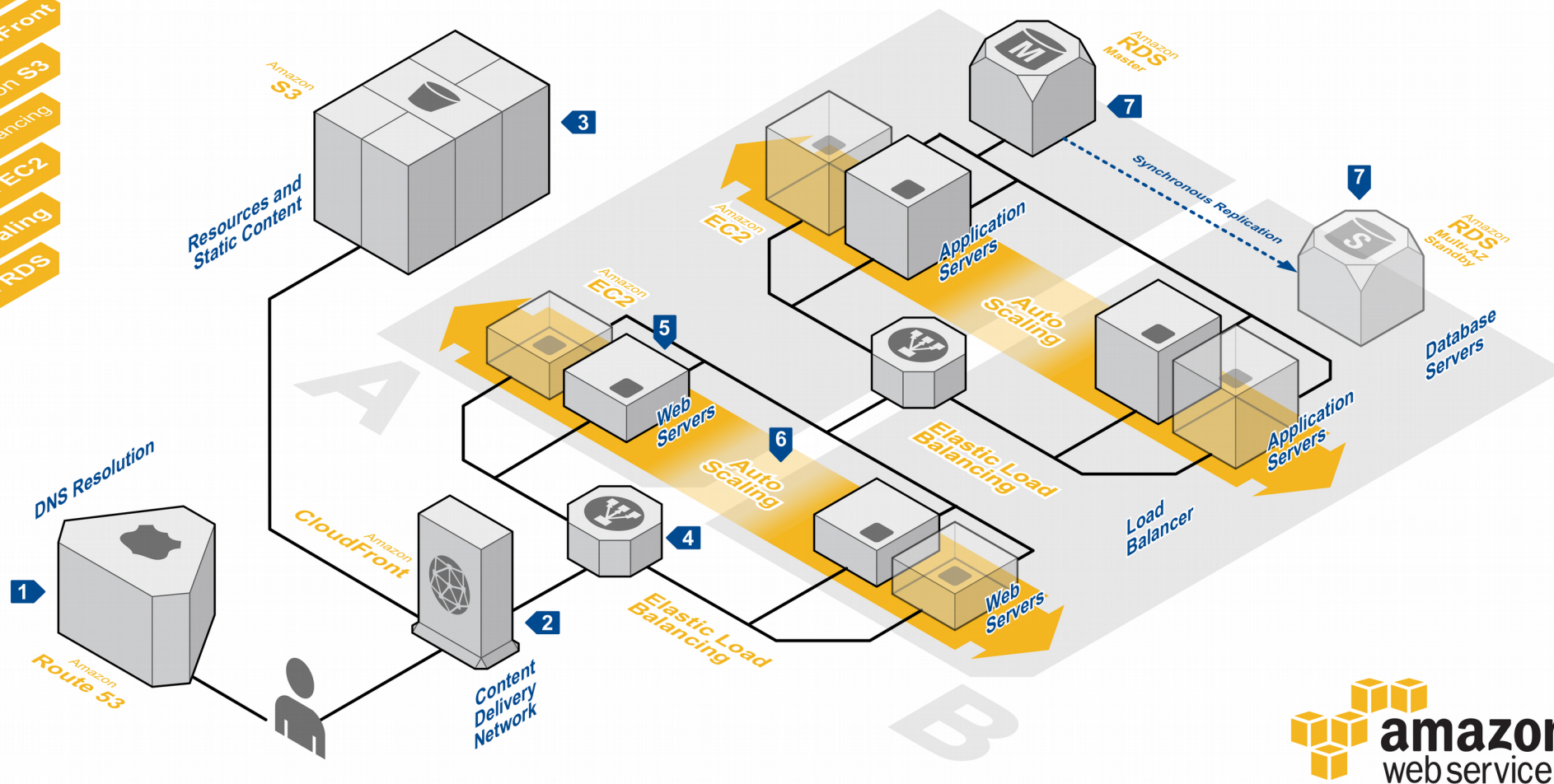
Consola AWS

Conectando todo lo visto: Arquitecturas de Referencia

<https://aws.amazon.com/architecture/>

WEB APPLICATION HOSTING

Highly available and scalable web hosting can be complex and expensive. Dense peak periods and wild swings in traffic patterns result in low utilization of expensive hardware. Amazon Web Services provides the reliable, scalable, secure, and high-performance infrastructure required for web applications while enabling an elastic, scale-out and scale-down infrastructure to match IT costs in real time as customer traffic fluctuates.



System Overview

- 1 The user's DNS requests are served by **Amazon Route 53**, a highly available Domain Name System (DNS) service. Network traffic is routed to infrastructure running in Amazon Web Services.
- 2 Static, streaming, and dynamic content is delivered by **Amazon CloudFront**, a global network of edge locations. Requests are automatically routed to the nearest edge location, so content is delivered with the best possible performance.
- 3 Resources and static content used by the web application are stored on **Amazon Simple Storage Service (S3)**, a highly durable storage infrastructure designed for mission-critical and primary data storage.

- 4 HTTP requests are first handled by **Elastic Load Balancing**, which automatically distributes incoming application traffic among multiple **Amazon Elastic Compute Cloud (EC2)** instances across Availability Zones (AZs). It enables even greater fault tolerance in your applications, seamlessly providing the amount of load balancing capacity needed in response to incoming application traffic.
- 5 Web servers and application servers are deployed on Amazon EC2 instances. Most organizations will select an **Amazon Machine Image (AMI)** and then customize it to their needs. This custom AMI will then become the starting point for future web development.

- 6 Web servers and application servers are deployed in an **Auto Scaling** group. Auto Scaling automatically adjusts your capacity up or down according to conditions you define. With Auto Scaling, you can ensure that the number of **Amazon EC2** instances you're using increases seamlessly during demand spikes to maintain performance and decreases automatically during demand to minimize costs.
- 7 To provide high availability, the relational database that contains application's data is hosted redundantly on a multi-AZ (multiple Availability Zones—zones A and B here) deployment of **Amazon Relational Database Service (Amazon RDS)**.

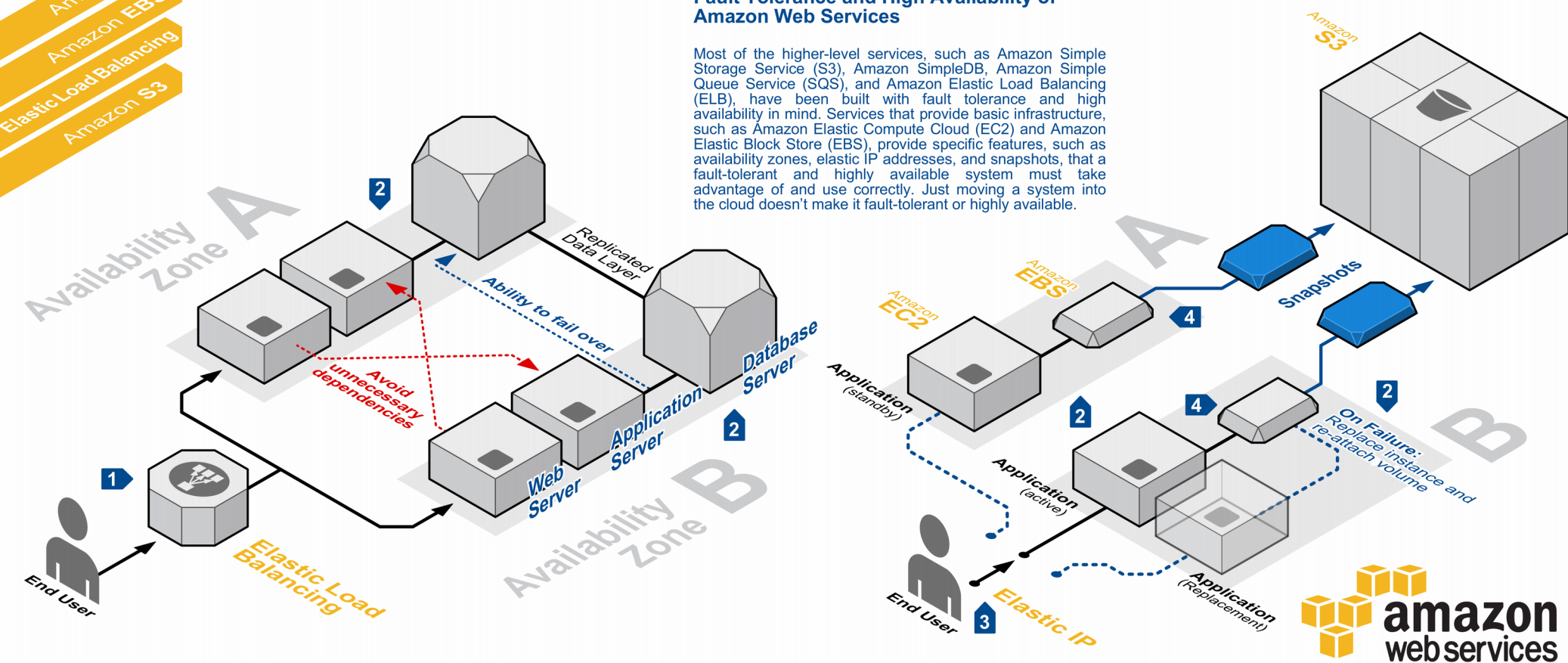
FAULT TOLERANCE & HIGH AVAILABILITY

Amazon Web Services provides services and infrastructure to build reliable, fault-tolerant, and highly available systems in the cloud. These qualities have been designed into our services both by handling such aspects without any special action by you and by providing features that must be used explicitly and correctly.

Amazon EC2 provides infrastructure building blocks that, by themselves, may not be fault-tolerant. Hard drives may fail, power supplies may fail, and racks may fail. It is important to use combinations of the features presented in this document to achieve fault tolerance and high availability.

Fault Tolerance and High Availability of Amazon Web Services

Most of the higher-level services, such as Amazon Simple Storage Service (S3), Amazon SimpleDB, Amazon Simple Queue Service (SQS), and Amazon Elastic Load Balancing (ELB), have been built with fault tolerance and high availability in mind. Services that provide basic infrastructure, such as Amazon Elastic Compute Cloud (EC2) and Amazon Elastic Block Store (EBS), provide specific features, such as availability zones, elastic IP addresses, and snapshots, that a fault-tolerant and highly available system must take advantage of and use correctly. Just moving a system into the cloud doesn't make it fault-tolerant or highly available.



System Overview

1 Load balancing is an effective way to increase the availability of a system. Instances that fail can be replaced seamlessly behind the load balancer while other instances continue to operate. **Elastic Load Balancing** can be used to balance across instances in multiple availability zones of a region.

2 **Availability zones (AZs)** are distinct geographical locations that are engineered to be insulated from failures in other AZs. By placing **Amazon EC2** instances in multiple AZs, an application can be protected from failure at a single location. It is important to run independent application stacks in more than one AZ, either in the same region or in another region, so that if one zone fails, the application in the other zone can continue to run. When you design such a

system, you will need a good understanding of zone dependencies.

3 **Elastic IP** addresses are public IP addresses that can be programmatically mapped between instances within a region. They are associated with the AWS account and not with a specific instance or lifetime of an instance. **Elastic IP** addresses can be used to work around host or availability zone failures by quickly remapping the address to another running instance or a replacement instance that was just started. Reserved instances can help guarantee that such capacity is available in another zone.

4 Valuable data should never be stored only on instance storage without proper backups, replication, or the

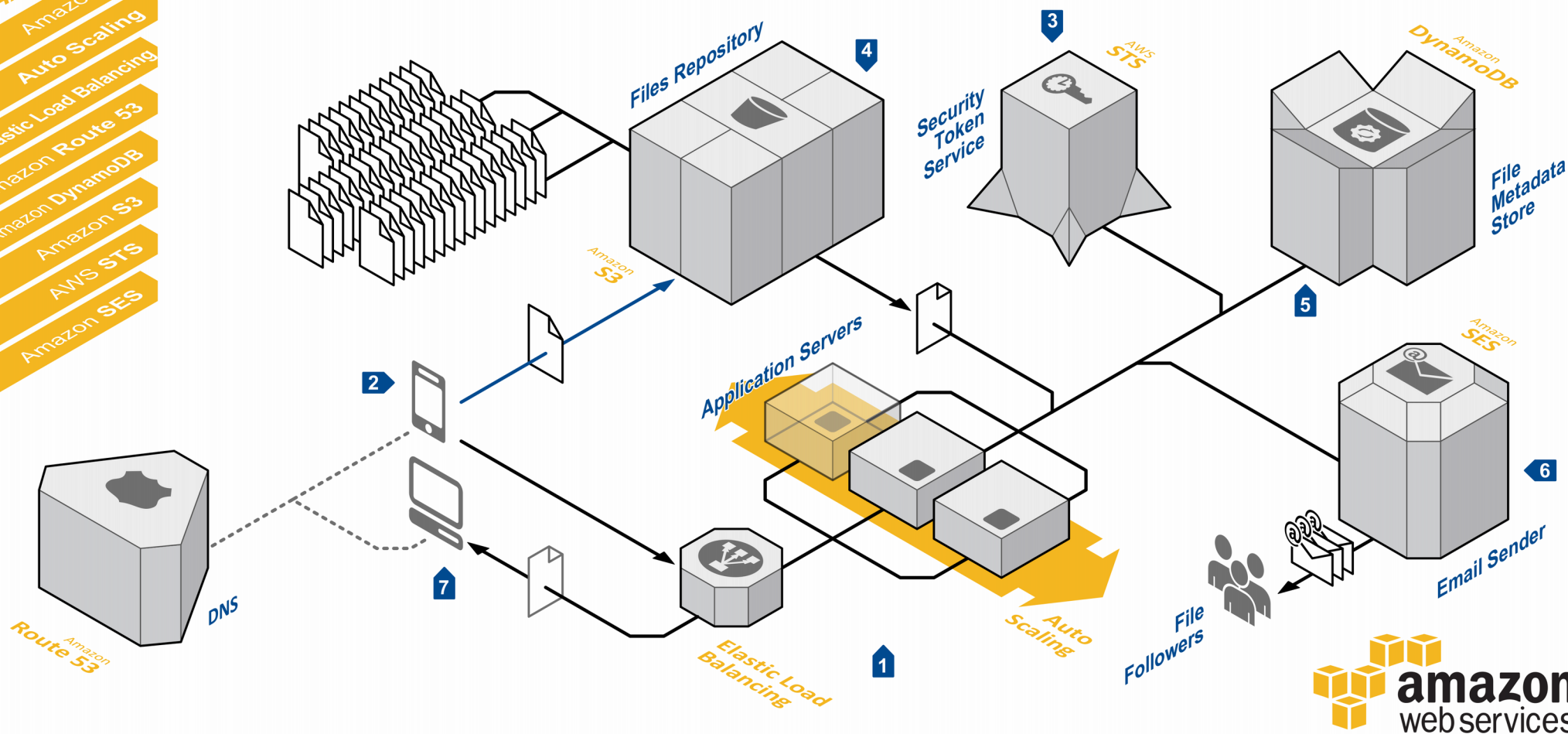
ability to re-create the data. **Amazon Elastic Block Store (EBS)** offers persistent off-instance storage volumes that are about an order of magnitude more durable than on-instance storage. EBS volumes are automatically replicated within a single availability zone. To increase durability further, point-in-time snapshots can be created to store data on volumes in **Amazon S3**, which is then replicated to multiple AZs. While EBS volumes are tied to a specific AZ, snapshots are tied to the region. Using a snapshot, you can create new EBS volumes in any of the AZs of the same region. This is an effective way to deal with disk failures or other host-level issues, as well as with problems affecting an AZ. Snapshots are incremental, so it is advisable to hold on to recent snapshots.



FILE SYNCHRONIZATION SERVICE

Given the straightforward, stateless client-server architecture in which web services are viewed as resources and can be identified by their URLs, development teams are free to create file sharing and syncing applications for their departments, for enterprises, or for consumers directly.

This diagram represents the core architecture of a scalable and cost-effective file sharing and synchronization platform, using Amazon Web Services.



System Overview

- 1 The file synchronization service endpoint consists of an **Elastic Load Balancer** distributing incoming requests to a group of application servers hosted on **Amazon Elastic Compute Cloud** (Amazon EC2) instances. An **Auto Scaling** group automatically adjusts the number of **Amazon EC2** instances depending on the application needs.
- 2 To upload a file, a client first needs to request the permission to the service and get a security token.
- 3 After checking the user's identity, application servers get a temporary credential from **AWS Security Token Service** (STS). This credential allows users to upload files.

- 4 Users upload files into **Amazon Simple Storage Service** (Amazon S3), a highly durable storage infrastructure designed for mission-critical and primary data storage. **Amazon S3** makes it easy to store and retrieve any amount of data, at any time. Large files can be uploaded by the same client using multiple concurrent threads to maximize bandwidth usage.

- 5 File metadata, version information, and unique identifiers are stored by the application servers on an **Amazon DynamoDB** table. As the number of files to maintain in the application grows, **Amazon DynamoDB** tables can store and retrieve any amount of data, and serve

any level of traffic.

- 6 File change notifications can be sent via email to users following the resource with **Amazon Simple Email Service** (Amazon SES), an easy-to-use, cost-effective email solution.

- 7 Other clients sharing the same files will query the service endpoint to check if newer versions are available. This query compares the list of local files checksums with the checksums listed in an **Amazon DynamoDB** table. If the query finds newer files, they can be retrieved from **Amazon S3** and sent to the client application.



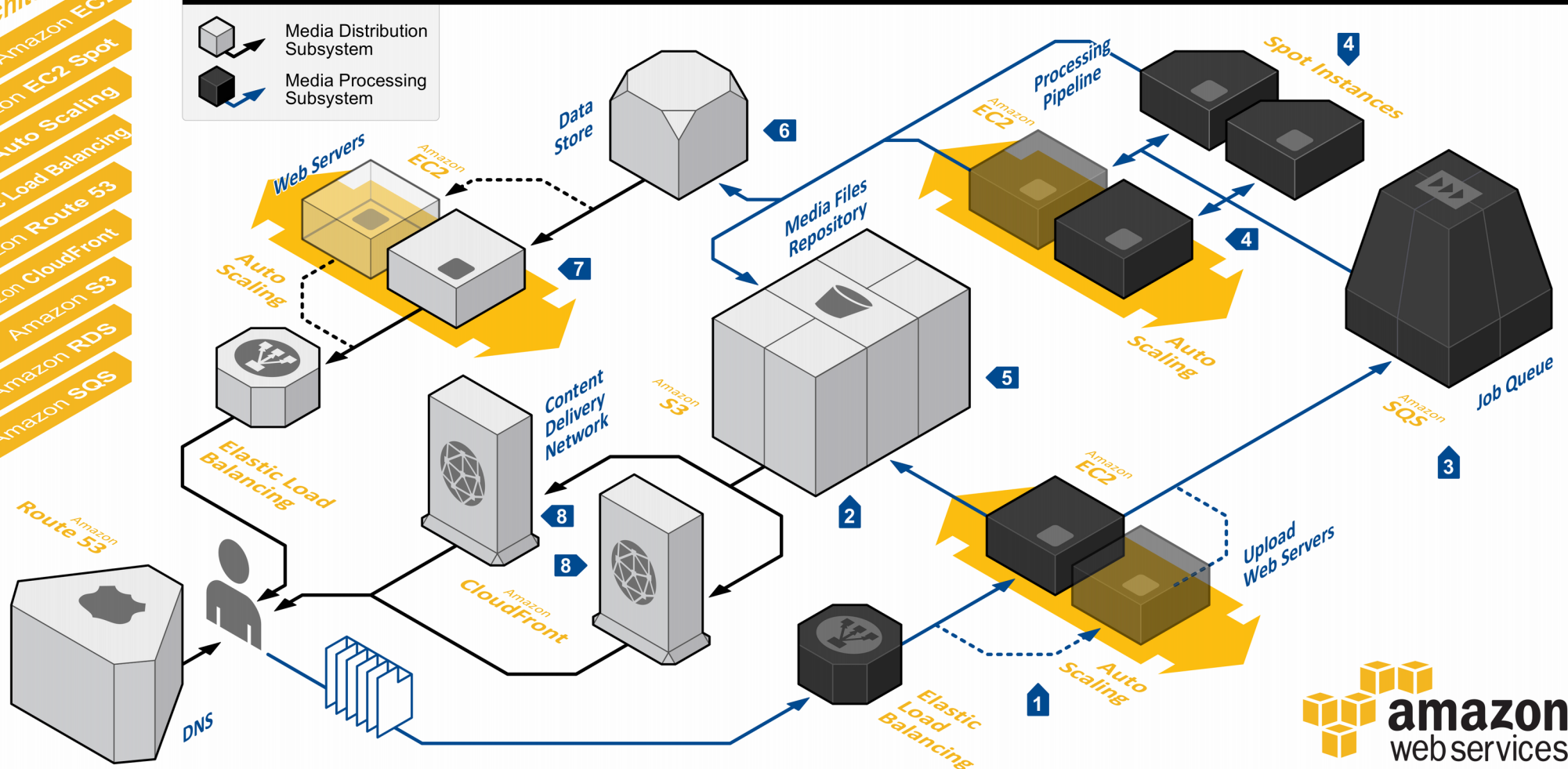
MEDIA SHARING

Media sharing is one of the hottest markets on the Internet. Customers have a staggering appetite for placing photos and videos on social networking sites, and for sharing their media in custom online photo albums.

The growing popularity of media sharing means scaling problems for site owners, who face ever-increasing storage and bandwidth requirements and increased go-to-market pressure to deliver faster than the competition.

Since most businesses today have limited manpower, budget, and data center space, AWS offers a unique set of opportunities to compete and scale without having to invest in hardware, staff, or additional data center space. Utilizing AWS is not an all or nothing proposition. Depending on the project, different services can be used independently.

This diagram shows an example of a highly available, durable, and cost-effective media sharing and processing platform.



System Overview

- 1 Sharing content first involves uploading media files to the online service. In this configuration, an **Elastic Load Balancer** distributes incoming network traffic to upload servers, a dynamic fleet of **Amazon Elastic Compute Cloud** (Amazon EC2) instances. **Amazon CloudWatch** monitors these servers and an **Auto Scaling** group manages them, automatically scaling EC2 capacity up or down based on load. In this example, a separate endpoint to receive media uploads was created in order to off-load this task from the website's servers.
- 2 Original uploaded files are stored in **Amazon Simple Storage Service** (Amazon S3), a highly available and durable storage service.

- 3 To submit a new file to be processed, upload web servers push a message into an **Amazon Simple Queue Service** (Amazon SQS) queue. This queue acts as a communication pipeline between the file reception and file processing components.
- 4 The processing pipeline is a dedicated group of post-processing task on the uploaded media files (video transcoding, image resizing, etc.). To automatically adjust the needed capacity, **Auto Scaling** manages this group. You can use **Spot Instances** to dynamically extend the capacity of the group and to significantly reduce the cost of file processing.

- 5 Once processing is completed, **Amazon S3** stores the output files. Original files can be stored with high durability. Processed files could use reduced redundancy.
- 6 Media-related data can be put in a relational database like **Amazon Relational Database Service** (Amazon RDS) or in a key-value store like **Amazon SimpleDB**.
- 7 A third fleet of EC2 instances is dedicated to host the website front-end of the media sharing service.
- 8 Media files are distributed from **Amazon S3** to the end user via **Amazon CloudFront**, a content delivery network. **Amazon CloudFront** offers low-latency delivery through a worldwide network of edge locations.