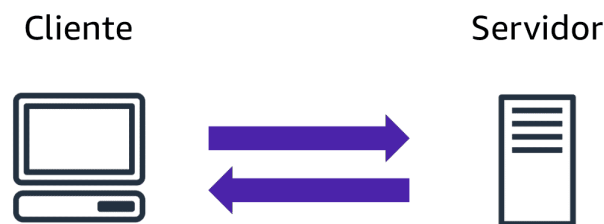


MÓDULO 1: Introducción a AWS y computación en la nube

1. Modelo Cliente-Servidor

El cliente hace solicitudes y el servidor las procesa. Ejemplo: al buscar algo en un navegador, el servidor (como EC2) devuelve la información solicitada.

Ejemplo: Cuando se busca algo en Google, el navegador es el cliente que envía la solicitud. Google tiene servidores (como los que AWS ofrece) que buscan la información y la envían.



2. Computación en la nube (AWS): Entrega de recursos de TI bajo demanda a través de Internet con pago por uso. Ejemplos: Obtener servidores virtuales o almacenamiento cuando se necesitan, pagando solo por lo usado.

3. Tipos de despliegue de la nube:

3.1. Nube: Todo en la nube, migración o creación de aplicaciones allí. Ejemplo: Guardar archivos en Google Drive (nada está en tu ordenador).

3.2. En las instalaciones: Recursos gestionados en los centros de datos de la empresa (nube privada). Ejemplo: Una empresa con servidores privados en su edificio.

3.3. Híbrido: Combina la nube con infraestructura local. Ejemplo: Una empresa usa la nube para almacenar copias de seguridad, pero tiene un servidor en sus oficinas para otras tareas.

4. Beneficios de la computación en la nube:

- **Gasto variable:** Pagar solo por lo que se usa.
- **Sin mantenimiento de centros de datos:** AWS se encarga del mantenimiento.

- **Escalabilidad:** Escalar según la demanda sin previsiones previas.
- **Economías de escala:** Precios más bajos gracias a la infraestructura global de AWS.
- **Velocidad y agilidad:** Desarrollo e implementación más rápidos.
- **Alcance global:** Expansión rápida y económica de infraestructura a nivel global.

5. Beneficios del modelo “pago por uso”:

Ayuda a la escala de la computación en la nube a ahorrar costes, y es más económico.

Permite pagar solo por lo que utilizas, sin necesidad de grandes inversiones iniciales. Esto es ideal para pruebas y desarrollo porque puedes ajustar recursos según las necesidades del proyecto, evitando gastos innecesarios.

MÓDULO 2: Amazon EC2

1. ¿Qué es Amazon EC2?: Servicio de AWS que proporciona servidores virtuales para tareas de computación. Los servidores físicos están gestionados por AWS.

2. Ventajas de EC2:

- **Flexible:** Inicia o detiene servidores cuando quieras.
- **Rápido:** Servidores listos en minutos.
- **Económico:** Pagas solo por lo que usas.
- **Sin complicaciones:** AWS gestiona los servidores y centros de datos.

3. Funcionamiento de EC2:

- **Virtualización:** Un servidor físico aloja varias instancias virtuales.
- **Hipervisor:** Controla los recursos entre las instancias y las aísla.
- **Personalización:** Elige el sistema operativo y configura las instancias.

4. Ventajas para empresas:

- **Escalabilidad:** Aumenta o reduce capacidad según sea necesario.
- **Rentabilidad:** Más económico que mantener servidores físicos.

5. Tipos de instancias EC2:

Las instancias de Amazon EC2 son servidores virtuales en la nube que permiten ejecutar aplicaciones.

1. **Instancias de propósito general:** Balance de recursos (computación, memoria, red). Para aplicaciones, juegos, bases de datos pequeñas.
En la cafetería: El cajero que atiende todo un poco.
2. **Instancias optimizadas para computación:** Alto rendimiento para tareas intensivas en cálculos. Para servidores web, procesamiento por lotes.
En la cafetería: El camarero rápido que solo lleva los pedidos.
3. **Instancias optimizadas para memoria:** Gran capacidad de memoria para bases de datos y análisis de datos en tiempo real.
En la cafetería: El cajero experto en atender a grupos grandes.
4. **Instancias de computación acelerada:** Usan hardware especializado (GPU, coprocesadores) para tareas gráficas y complejas.
En la cafetería: El barista experto en preparar bebidas especiales.
5. **Instancias optimizadas para almacenamiento:** Alta velocidad y baja latencia para grandes volúmenes de datos. Para sistemas de archivos distribuidos y procesamiento de transacciones.
En la cafetería: El almacén donde se guardan todos los ingredientes.

6. Precios de Amazon EC2

- **Bajo demanda:** Pagas solo por uso (ideal para pruebas).
Ejemplo: Usar un servidor solo unas horas.
- **Savings Plans:** Compromiso por 1-3 años, ahorros de hasta 72%.
Ejemplo: si sabes que usarás EC2 regularmente.
- **Instancias reservadas:** Ahorro de hasta 75% con compromiso por 1-3 años.
Ejemplo: Pagas por adelantado para ahorrar.
- **Instancias Spot:** Ahorro de hasta 90%, pero pueden ser interrumpidas.
Ejemplo: Perfectas para tareas no urgentes.
- **Hosts dedicados:** Servidores físicos exclusivos para un cliente.
Ejemplo: Para empresas con necesidades especiales (como seguridad).

7. Escalabilidad y elasticidad

7.1. Auto Scaling: Ajusta automáticamente el número de instancias según demanda.
Ejemplo: Si tu web recibe muchos visitantes, se crean más instancias automáticamente para atender a todos.

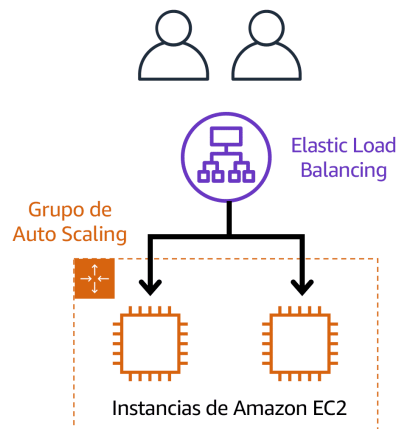
7.2. Escalado horizontal: Añadir más instancias. Ejemplo: Contratar más empleados en la cafetería.

7.3. Escalado vertical: Aumentar la capacidad de una instancia. Ejemplo: Mejorar el equipo del cajero para que atienda más rápido.

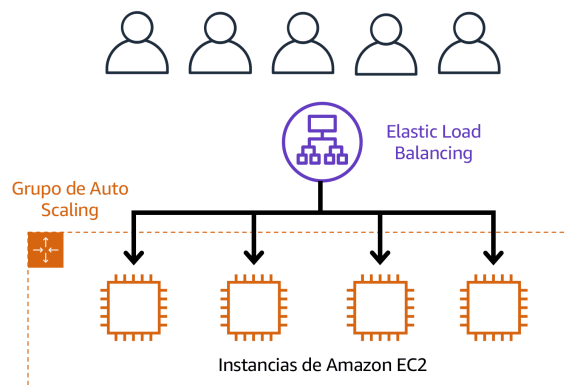
8. Elastic Load Balancing (ELB)

Distribuye el tráfico entre instancias EC2 para evitar sobrecarga. Se ajusta automáticamente al tráfico y trabaja con Auto Scaling. Ejemplo: Como un empleado que dirige a los clientes a la caja que tenga menos fila.

8.1. Periodo de baja demanda: Imagina que solo hay pocos clientes en la cafetería. Solo se abren las cajas registradoras necesarias (instancias de EC2), evitando tener cajas abiertas sin clientes.



8.2. Periodo de alta demanda: A medida que llegan más clientes, se abren más cajas registradoras para atenderlos. Un empleado (el balanceador de carga) dirige a los clientes a la caja adecuada para repartir los pedidos de manera uniforme.



9. Mensajería y colas

9.1. Amazon SQS: gestiona la cola de mensajes de manera eficiente y fiable. Almacena y envía mensajes entre componentes de Software. **Ejemplo:** Si un cliente pide algo y el camarero tarda, la orden queda en una lista hasta que sea atendida.

9.2. Amazon SNS: difunde mensajes a múltiples suscriptores, incluidos usuarios finales).

Ejemplo: El barista manda una notificación a todos los empleados cuando está lista una nueva bebida.

Ambos servicios permiten una arquitectura de acoplamiento débil, asegurando que los sistemas sigan funcionando incluso cuando algunos componentes fallan.

Microservicios: Componentes independientes que se comunican mediante colas o mensajes.

10. Servicios de computación adicionales

AWS (Amazon Web Services)

Ofrece servicios para crear soluciones personalizadas, incluyendo:

- **EC2 (Elastic Compute Cloud):** Servidores virtuales escalables. Modelos de precios: bajo demanda, spot, instancias reservadas y Savings Plans.
- **Elastic Load Balancer:** Distribuye tráfico entre instancias EC2.
- **SQS (Simple Queue Service):** Desacopla componentes mediante colas de mensajes.
- **SNS (Simple Notification Service):** Envía mensajes a suscriptores (correo, SMS, etc.).
- **Servicios de contenedores:**
 - **ECS y EKS:** Para manejar contenedores.
 - **Fargate:** Ejecuta contenedores sin servidor.
 - **Lambda:** Para aplicaciones con ejecución rápida sin gestionar servidores.

MÓDULO 3: Infraestructura global y fiabilidad.

1. Infraestructura global de AWS:

AWS cuenta con una infraestructura distribuida por todo el mundo para garantizar alta disponibilidad, baja latencia y cumplimiento normativo.

1.1. Regiones: Áreas geográficas aisladas con centros de datos, conectadas por fibra óptica de alta velocidad. Las regiones están interconectadas mediante fibra óptica de alta velocidad.

1.2. Alta disponibilidad: AWS replica la infraestructura para mantener servicios operativos ante fallos locales.

1.3. Factores para elegir región:

1. **Conformidad:** Cumple con regulaciones locales (ej. datos en España).
2. **Proximidad:** Elige una región cerca de tus clientes para reducir la latencia.
3. **Servicios disponibles:** Algunas regiones tienen servicios específicos.
4. **Precios:** Los costos varían entre regiones, influenciados por impuestos y otros factores.

2. Zonas de disponibilidad (AZ)

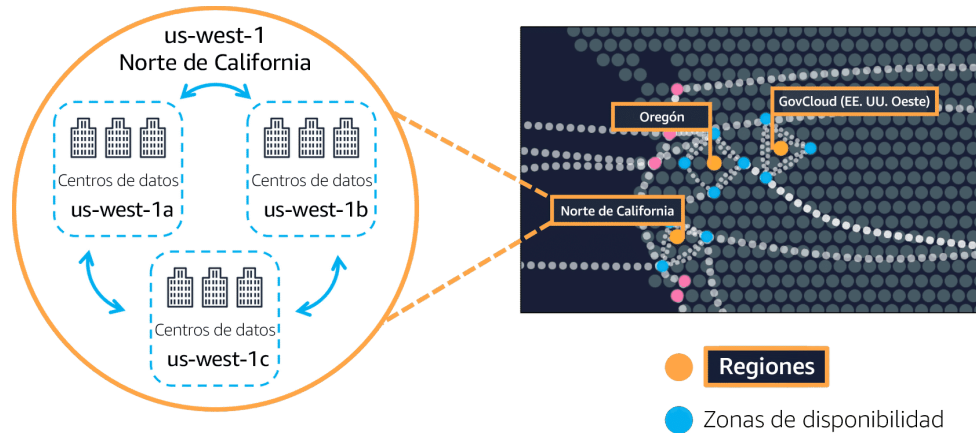
2.1. Una región contiene varias zonas de disponibilidad (AZ), que son centros de datos independientes pero interconectados.

AWS tiene **regiones, zonas de disponibilidad (AZ)** y **ubicaciones periféricas** distribuidas por todo el mundo para ofrecer servicios rápidos y confiables.

Es recomendable usar al menos dos **zonas de disponibilidad** para mayor seguridad y rendimiento. Cada **zona de disponibilidad** está formada por centros de datos separados por decenas de kilómetros, lo que reduce el riesgo de fallos simultáneos.

2.2. Beneficios:

- **Redundancia:** Usar mínimo 2 AZ para garantizar disponibilidad.
- **Baja latencia:** Menos de 10 ms entre AZ.
- **Alta disponibilidad:** Servicios como ELB distribuyen automáticamente el tráfico.



3. Ubicaciones periféricas: Centros diseñados para mejorar la entrega de contenido a usuarios lejanos de las regiones principales.

3.1. CloudFront: CDN que entrega contenido rápido y con baja latencia desde ubicaciones cercanas.

3.2. Beneficios:

- **Baja latencia:** Mejora la velocidad de entrega de contenido.
- **Independencia regional:** CloudFront funciona globalmente, no depende de una región específica.

3.3. Route 53

Es un servicio de DNS que redirige el tráfico a las ubicaciones más cercanas para mejorar el rendimiento.

3.4. AWS Outposts: AWS puede instalar una mini región operativa en las instalaciones, brindando toda la funcionalidad de AWS localmente.

3.5. Ejemplo:

Problema: Si tus clientes están lejos de una región de AWS, acceder a tus datos puede ser lento.

Solución: AWS usa ubicaciones periféricas y **Amazon CloudFront** para almacenar copias de los datos más cerca de los usuarios, garantizando baja latencia y alta velocidad de entrega.

4. Aprovisionamiento de recursos en AWS

4.1. Interacción con AWS:

- **Consola de administración:** Es una interfaz visual basada en navegador, ideal para principiantes o tareas puntuales. Permite gestionar recursos de manera intuitiva, ideal para entornos de prueba, facturación y control visual.
- **AWS CLI:** Permite automatizar y realizar tareas mediante la línea de comandos. Ideal para crear y gestionar recursos repetidamente a través de scripts, lo que minimiza errores humanos y optimiza el proceso de gestión.
- **SDKs:** Integra AWS en aplicaciones (Java, Python, etc.). Facilitan la creación de aplicaciones a gran escala con servicios de AWS, sin necesidad de interactuar manualmente con la API de AWS.

4.2. Herramientas avanzadas:

- **AWS Elastic Beanstalk:** Gestiona automáticamente la infraestructura. Perfecto para desarrolladores que desean enfocarse en su código sin preocuparse por la infraestructura.
- **AWS CloudFormation:** Aprovisiona recursos con código (JSON/YAML). Ideal para equipos que necesitan desplegar la misma infraestructura en múltiples regiones o cuentas, asegurando consistencia y control de versiones.
- **Amazon CloudFront:** es una CDN (Red de Entrega de Contenido) que acelera la entrega de contenido web y aplicaciones a usuarios globalmente, mejorando la latencia y disponibilidad, y permite manejar grandes volúmenes de tráfico de manera eficiente y segura.

4.3. Ventajas de la automatización:

- Reducción de errores humanos.
- Capacidad para escalar rápidamente, creando y eliminando recursos según la demanda.
- Gestión eficiente de infraestructura, liberando a los equipos de tareas repetitivas y permitiendo que se enfoquen en el desarrollo y la innovación.

MÓDULO 4: Redes.

1. Conectividad a AWS

Amazon VPC y sus herramientas (subredes, puertas de enlace y Direct Connect) permiten crear redes seguras, flexibles y eficientes para gestionar tus recursos en AWS.

Todo está diseñado para adaptarse a las necesidades específicas de cada cliente, ya sea acceso público o conexiones privadas exclusivas.

Amazon VPC facilita la creación de redes seguras al proporcionar un entorno aislado donde puedes controlar el tráfico mediante grupos de seguridad, NACLs y tablas de enrutamiento.

- **Amazon VPC (Virtual Private Cloud):** Red privada en AWS, dividida en subredes:
 - **Subred pública:** Con acceso a Internet.
 - **Subred privada:** Sin acceso a Internet.
- **Puerta de Enlace de Internet (IGW):** Permite el acceso público a los recursos en la VPC.
- **Puerta de Enlace Privada Virtual (VGW):** Crea una conexión segura entre la red privada (empresa) y la VPC mediante VPN.
- **AWS Direct Connect:** Ofrece conexión privada dedicada, con baja latencia, entre tu centro de datos y la VPC, sin depender de Internet.

1.1. ¿Qué ocurre si tienes una VPC que incluye solo recursos privados?

Puedes usar una **puerta de enlace privada virtual** para conectarte de forma segura.

Es como usar una carretera pública (Internet) con un guardaespaldas (VPN) que protege tus datos. Aunque estás protegido, puedes experimentar atascos porque compartes la carretera con otros usuarios. La puerta de enlace privada virtual permite que el tráfico cifrado entre en la VPC de forma segura.

2. Subredes y listas de control de acceso a la red

2.1. VPC (Virtual Private Cloud):

Es como una fortaleza que controla el tráfico que entra o sale. La VPC tiene herramientas de seguridad a diferentes niveles como:

- Seguridad de redes.
- Control de identidad y accesos.
- Protección contra ataques DDoS.
- Cifrado de datos.

Herramientas clave:

- **Puerta de Enlace de Internet (IGW):** Conexión al mundo exterior.
- **Puerta de Enlace Virtual Privada (VGW):** Conexión con redes privadas.

2.2. Subredes:

- **Subred pública:** Aloja recursos accesibles desde Internet (ej. servidores web en instancias EC2).
- **Subred privada:** Aloja recursos internos no accesibles desde Internet (ej. bases de datos).

2.3. ACL de Red (Listas de Control de Acceso): Actúan como agentes de control de pasaportes en el límite de las subredes.

Reglas predeterminadas:

- **ACL predeterminada:** Permite todo el tráfico.
- **ACL personalizada:** Bloquea todo por defecto hasta que configures reglas específicas.

2.4. Grupos de seguridad: Son firewalls asociados a las instancias de EC2.

- Controlan el tráfico **entrante** y **saliente** de cada instancia.
- Son **con estado**, lo que significa que recuerdan el tráfico previamente aprobado.

Reglas predeterminadas:

- **Tráfico entrante:** Todo está bloqueado.
- **Tráfico saliente:** Todo está permitido.

2.5. Diferencias clave entre ACLs y grupos de seguridad:

- **ACL:** Controla el tráfico a nivel de subred, es **sin estado** y verifica cada paquete.
- **Grupo de Seguridad:** Controla el tráfico a nivel de instancia EC2, es **con estado** y recuerda las conexiones previas.

2.6. Analogías:

- **VPC:** Una fortaleza.
- **Subred:** Zonas separadas dentro de la fortaleza (pública o privada).
- **ACL:** Agente de control de pasaportes que revisa cada entrada y salida sin importar el estado de la conexión.
- **Grupo de seguridad:** Un portero que verifica la entrada y salida y recuerda quién fue aprobado.

3. Redes globales.

3.1. DNS (Sistema de Nombres de Dominio): Traduce nombres de dominio a direcciones IP.

Es como una guía telefónica de Internet: traduce nombres de dominio (como anycompany.com) a direcciones IP que los ordenadores pueden entender.

Cuando escribes una dirección web, tu navegador pregunta a un servicio DNS por la dirección IP del sitio web.

3.2. Amazon Route 53: es un servicio de DNS escalable que no solo traduce nombres de dominio a direcciones IP, sino que también permite el enrutamiento de tráfico con políticas avanzadas, como enrutamiento basado en latencia y geolocalización.

3.3. Amazon CloudFront: Es una red de entrega de contenido (CDN) globalmente distribuida que acelera la entrega de contenido al distribuirlo desde ubicaciones periféricas cercanas al usuario final. Esto reduce la latencia y mejora el tiempo de carga.

3.4. Cómo trabajan juntos Route 53 y CloudFront:

Route 53 resuelve el nombre de dominio, mientras que **CloudFront** entrega el contenido desde la ubicación más cercana, optimizando la entrega y reduciendo la latencia. Si el contenido está alojado en **instancias EC2**, el tráfico es dirigido a un **Application Load Balancer**, que distribuye las solicitudes a las instancias EC2 correspondientes.

MÓDULO 5: Almacenamiento y Bases de Datos

1. Almacenamiento en AWS para bases de datos y aplicaciones

1.1. Almacenamiento en bloques: Almacena datos en bloques que se pueden actualizar individualmente. Ideal para bases de datos y sistemas de archivos. Ejemplo: Disco duro o SSD de un ordenador.

1.2. Opciones de almacenamiento en EC2:

- **Almacén de Instancias:** Almacenamiento temporal en el host de la instancia EC2, útil para datos efímeros. Los datos se pierden al detenerse la instancia.
- **Amazon EBS:** Almacenamiento persistente independiente del host físico, ideal para datos importantes. Permite copias de seguridad incrementales.

1.3. Instantáneas de EBS: Copias incrementales de volúmenes de EBS. Ahorran espacio y tiempo, protegen datos y permiten restaurarlos en caso de fallo.

2. Amazon Simple Storage Service (Amazon S3)

Es un servicio de almacenamiento en la nube. Almacena objetos en buckets. Tamaño máximo de objeto: 5 TB.

2.1. Características:

- **Control de versiones:** Protege contra eliminaciones accidentales.
- **Permisos:** Control de acceso a los objetos.
- **Políticas de ciclo de vida:** Automatiza el movimiento de datos.

2.2. Clases de almacenamiento: Varían en costo y velocidad de recuperación.

- **S3 Standard:** Datos de acceso frecuente, alta disponibilidad.
- **S3 Standard-IA:** Datos de acceso infrecuente, bajo costo de almacenamiento, mayor costo por acceso.
- **S3 One Zone-IA:** Datos infrecuentes, almacenados en una sola zona, a menor costo.
- **S3 Glacier:** Almacenamiento a largo plazo, baja disponibilidad, bajo costo.

2.3. Ventajas: Escalabilidad ilimitada, flexibilidad, seguridad y costos ajustados.

3. Amazon Elastic File System (Amazon EFS)

Es un sistema de archivos compartido y escalable, ideal para aplicaciones con acceso concurrente desde múltiples instancias EC2.

3.1. Diferencias con EFS, EBS y S3: EFS permite acceso concurrente y se escala automáticamente, mientras que EBS no lo permite y requiere definir su tamaño. S3 es más para almacenamiento masivo de objetos.

3.2. Comparación entre Amazon EBS y Amazon EFS

Amazon EBS guarda datos en una sola zona de disponibilidad y solo puede usarse con instancias EC2 en esa misma zona.

Amazon EFS guarda datos en varias zonas de disponibilidad a la vez, lo que permite acceder a los datos desde cualquier zona de la región. Además, se puede acceder desde servidores fuera de AWS mediante AWS Direct Connect.

3.3. Ventajas: Escalabilidad automática, alta disponibilidad, gestión simplificada.

4. Amazon Relational Database Service (Amazon RDS)

Es un servicio administrado para bases de datos relacionales (MySQL, PostgreSQL, MariaDB, Oracle, SQL Server, Aurora). Automatiza tareas administrativas.

4.1. Ventajas: Cifrado, escalabilidad, recuperación ante desastres.

Comparativa con Aurora: Aurora es más rápida y ofrece alta disponibilidad.

5. Amazon DynamoDB: Base de datos NoSQL, sin servidor, clave-valor. Ofrece alta disponibilidad, rendimiento de milisegundos y escalabilidad automática.

5.1. Diferencias con RDS: DynamoDB no usa SQL, es ideal para cargas de trabajo con datos flexibles y sin relaciones complejas.

5.2. Casos de uso: Aplicaciones de alta demanda (e-commerce, IoT).

6. Amazon Redshift: Servicio de almacenamiento de datos optimizado para análisis de big data.

6.1. Ventajas: Capacidad de manejar petabytes de datos, alto rendimiento, ideal para Business Intelligence (BI) y análisis históricos.

6.2. Comparativa con bases tradicionales: Redshift es más adecuado para análisis de grandes volúmenes de datos históricos, mientras que bases tradicionales son mejores para transacciones en tiempo real.

7. AWS Database Migration Service (DMS): Facilita la migración de bases de datos a AWS, minimizando el tiempo de inactividad.

7.1. Tipos de migración:

- **Homogénea:** Entre bases de datos del mismo tipo.
- **Heterogénea:** Entre bases de datos de diferentes tipos, usando AWS Schema Conversion Tool.

7.2. Usos: Migración, replicación de datos, consolidación de bases de datos.

7.3. Beneficios: Mínimo tiempo de inactividad, compatibilidad entre bases de datos, fácil de usar.

8. Otros Servicios de Bases de Datos en AWS

- **Amazon DocumentDB:** Base de datos de documentos compatible con MongoDB.
- **Amazon Neptune:** Base de datos de grafos, ideal para relaciones complejas.
- **Amazon QLDB:** Base de datos de libro mayor para registros inmutables.
- **Amazon Managed Blockchain:** Para gestionar redes de blockchain.
- **Amazon ElastiCache:** Mejora la velocidad de las lecturas en bases de datos mediante caché en memoria (Redis, Memcached).
- **Amazon DAX:** Caché en memoria para DynamoDB, mejora la latencia de lectura.

Conclusión: AWS ofrece una variedad de servicios de almacenamiento y bases de datos, cada uno adecuado a diferentes necesidades de rendimiento, escalabilidad, y tipo de datos. Elegir el servicio adecuado depende de los requisitos específicos de la aplicación.