



Integración de microservicios mediante el uso de servicios AWS sin servidor

# AWS Orientación prescriptiva



# AWS Orientación prescriptiva: Integración de microservicios mediante el uso de servicios AWS sin servidor

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Las marcas comerciales y la imagen comercial de Amazon no se pueden utilizar en relación con ningún producto o servicio que no sea de Amazon, de ninguna manera que pueda causar confusión entre los clientes y que menosprecie o desacredite a Amazon. Todas las demás marcas registradas que no son propiedad de Amazon son propiedad de sus respectivos propietarios, que pueden o no estar afiliados, conectados o patrocinados por Amazon.

---

# Table of Contents

Introducción .....	1
Destinatarios previstos .....	2
Objetivos .....	2
Seguridad .....	2
Patrones de comunicación .....	3
Comunicación síncrona .....	3
Comunicación asíncrona .....	5
Disparo y olvido .....	6
Verificación de reclamación .....	6
Devolución de llamada .....	7
Comunicación bidireccional .....	8
Opciones de coordinación .....	11
Orquestación .....	11
Ejemplo: Step Functions .....	11
Ejemplo: Amazon MWAA .....	14
Diferencias clave entre Step Functions y Amazon MWAA .....	16
Coreografía .....	16
Elección de su enfoque de coordinación .....	17
Administrar APIs .....	18
Amazon API Gateway .....	18
Autenticación y autorización .....	18
Claves de API y límites de velocidad .....	19
Público y privado APIs .....	19
Cuándo usar API Gateway .....	20
Mensajería .....	21
Amazon SQS .....	21
Sondeo .....	22
Directrices .....	22
Amazon SNS .....	22
Directrices .....	24
Amazon EventBridge .....	24
Directrices .....	25
AWS AppSync Events y API Gateway .....	26
Directrices .....	26

Preguntas frecuentes .....	27
¿Cómo puedo combinar diferentes patrones de integración? .....	27
¿Cuál es la principal ventaja de utilizar una arquitectura de microservicios? .....	27
¿Cómo puedo implementar la gestión de errores? .....	27
¿Cuáles son los beneficios de usar el patrón de verificación de reclamaciones? .....	27
¿Cuáles son los beneficios principales del patrón de devolución de llamadas? .....	28
¿Puedo implementar la comunicación bidireccional? .....	28
¿Cómo puedo optimizar el uso de las funciones de Lambda? .....	28
¿Cuáles son las principales diferencias entre el uso de Amazon SNS y? EventBridge .....	28
Recursos .....	29
Servicio de AWS Documentación de .....	29
Lecturas adicionales .....	29
Historial de documentos .....	30
Glosario .....	31
# .....	31
A .....	32
B .....	35
C .....	37
D .....	41
E .....	45
F .....	47
G .....	49
H .....	51
I .....	52
L .....	55
M .....	56
O .....	60
P .....	63
Q .....	66
R .....	67
S .....	70
T .....	74
U .....	76
V .....	76
W .....	77
Z .....	78



# Integración de microservicios mediante servicios AWS sin servidor

Tabby Ward, Abhishek Agawane y Matt Kahn, Amazon Web Services

Septiembre [de](#) 2025 (historial del documento)

Un aspecto importante de la modernización del software de la organización es la selección de los patrones de arquitectura adecuados que permitan la agilidad y la capacidad de respuesta ante las cambiantes necesidades empresariales. En algunas aplicaciones, una arquitectura monolítica es una opción habitual. Sin embargo, para muchas organizaciones, la [refactorización de los monolitos en microservicios](#) puede ser una estrategia de modernización eficaz cuando el caso de uso se alinea con las ventajas de los microservicios.

Los microservicios y los monolitos no se excluyen mutuamente: muchas organizaciones exitosas utilizan ambos de forma conjunta, en los que los monolitos modulares sirven para algunos dominios y los microservicios para otros.

Cuando los microservicios son parte de la arquitectura, se puede recurrir a varios servicios para obtener datos para una transacción empresarial. La implementación de estas integraciones requiere un diseño cuidadoso para abordar posibles desafíos como la coherencia de datos, la latencia y la complejidad operativa. Cuando los microservicios se integran correctamente, pueden ofrecer beneficios como un escalado independiente, una mayor velocidad de desarrollo y posibles optimizaciones de costos.

La guía forma parte de una serie de contenido que cubre el enfoque de modernización de aplicaciones recomendado por AWS. La serie también incluye:

- [Estrategia para modernizar las aplicaciones en el Nube de AWS](#)
- [Enfoque gradual para modernizar las aplicaciones en el Nube de AWS](#)
- [Evaluación de la preparación para la modernización de las aplicaciones en la nube de Nube de AWS](#)
- [Descomposición de monolitos en microservicios](#)

## Destinatarios previstos

Esta guía es útil para propietarios de aplicaciones, de empresas, arquitectos, líderes técnicos y gerentes de proyectos, que han determinado que los microservicios son adecuados para su caso de uso específico. La guía presenta varios patrones de comunicación sincrónica y asíncrona entre microservicios mediante el uso de sistemas sin servidor, como Servicios de AWS Amazon API AWS Lambda Gateway, para lograr autonomía y escalabilidad.

## Objetivos

Al usar esta guía para integrar sus nuevos microservicios, puede transformar de manera eficiente la arquitectura de su organización en una arquitectura de microservicios. Esto ayuda a proporcionar un ajuste rápido a las necesidades comerciales fluctuantes mediante alta escalabilidad, resiliencia mejorada, entrega continua y aislamiento de fallos. Una arquitectura de microservicios también acelera la innovación, ya que cada microservicio se puede implementar y probar de forma individual.

Una arquitectura de microservicios también puede ayudar a reducir el tiempo de comercialización de sus productos o servicios, ya que cada microservicio tiene una base de código independiente que facilita y agiliza la adición de nuevas características y su iteración.

## Seguridad

Debe proteger sus microservicios de manera adecuada para proteger la integridad de sus servicios y datos, pero debe asegurarse de que la seguridad no afecte de forma negativa al rendimiento de la aplicación.

En un entorno de microservicios, debe tener en cuenta cómo cada servicio autenticará y autorizará las solicitudes que reciba de un cliente externo o de otro microservicio. Tenga en cuenta también cómo cada servicio accederá de forma segura a los demás Servicios de AWS.

[El acceso a Servicios de AWS debe concederse a través de funciones de ámbito limitado \(IAM\).AWS Identity and Access Management](#) Al asumir un rol de IAM, el microservicio dispone de credenciales de IAM a corto plazo en forma de clave de acceso, secreto de acceso y token de sesión. Los distintos kits de desarrollo de software (SDKs) los utilizan para firmar las solicitudes Servicios de AWS mediante la [versión 4 de AWS Signature](#) (SiGv4).

# Patrones de comunicación

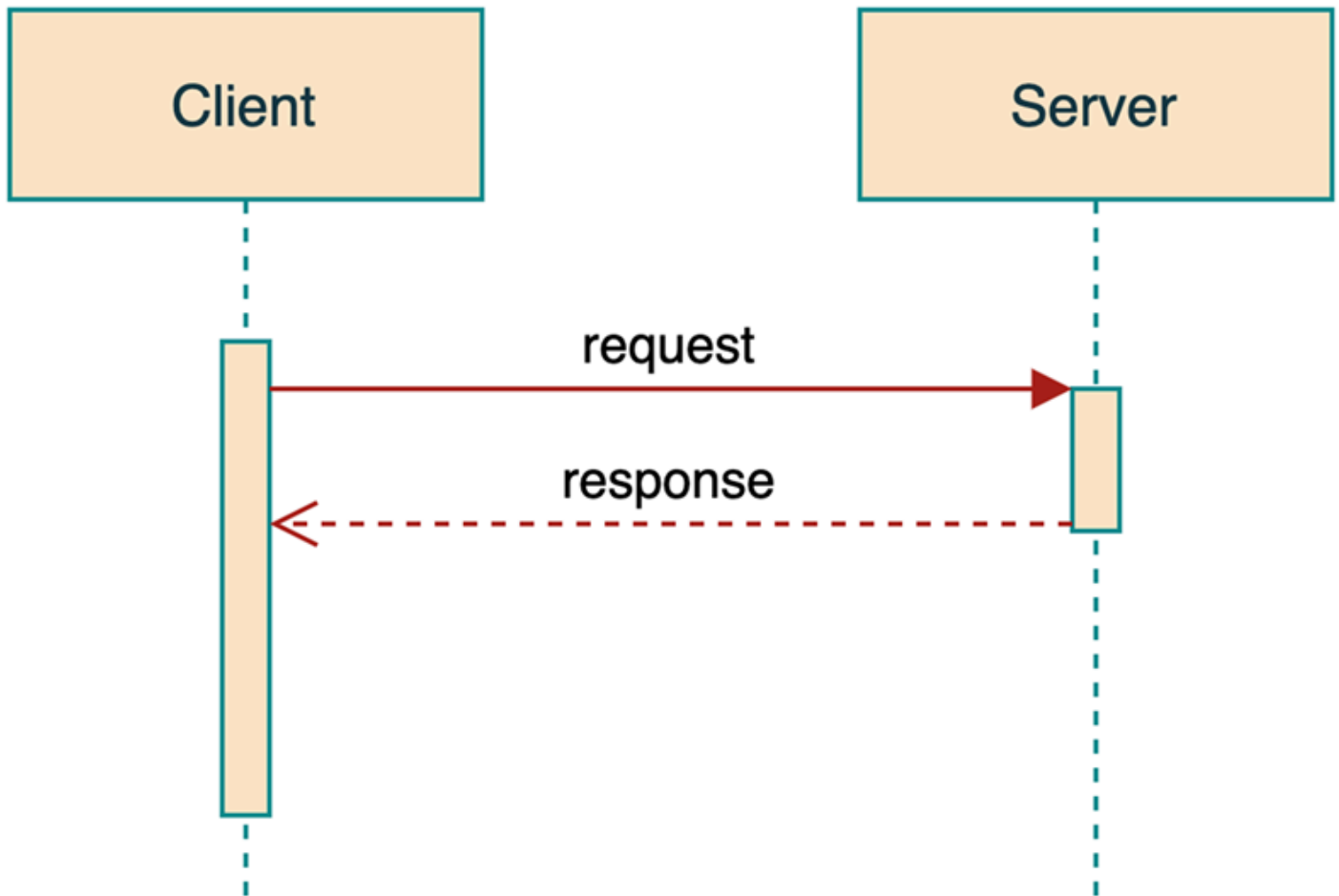
En una arquitectura de microservicios, la comunicación tiene lugar en dos patrones, síncronos y asíncronos. En la comunicación síncrona, la persona que llama espera una respuesta antes de continuar, de forma similar a una llamada a la API de REST HTTP en tiempo real. La comunicación asíncrona sigue un patrón basado en mensajes en el que la persona que llama continúa el procesamiento sin esperar una respuesta, por ejemplo, cuando se utilizan colas de mensajes. En las siguientes secciones se examinan en detalle la implementación, las ventajas y los casos de uso de cada patrón.

## Temas

- [Comunicación síncrona](#)
- [Comunicación asíncrona](#)

## Comunicación síncrona

En la comunicación síncrona, un cliente inicia una solicitud a un servicio, como se muestra en el siguiente diagrama. Los ejemplos incluyen una solicitud para obtener información, como una solicitud HTTP GET, o una solicitud para mutar datos, como una solicitud HTTP PUT. En cualquier caso, el cliente espera a que el servidor responda antes de continuar. La mayoría de los desarrolladores conocen las llamadas síncronas, son fáciles de implementar y solucionar problemas y, en muchos casos, son el estándar de comunicación ampliamente aceptado.



Los beneficios de la comunicación síncrona incluyen:

- Control de flujo predecible: ejecución determinista y ciclos claros de solicitud-respuesta que son más fáciles de entender en comparación con la comunicación asíncrona.
- Consistencia sólida: confirmación inmediata de los cambios en los datos y las actualizaciones de estado.
- Gestión sencilla de errores: propagación directa de errores y excepciones.
- Depuración sencilla: seguimiento y supervisión sencillos de las solicitudes.
- Soporte de protocolos: protocolos bien establecidos, como HTTP y REST, que facilitan la implementación.

La comunicación síncrona tiene algunos inconvenientes:

- Acoplamiento ajustado: los servicios dependen directamente de la disponibilidad de los demás.

- Impacto en la red: aumento de la carga de la red debido a las conexiones abiertas constantes.
- Utilización de recursos: mayor uso de memoria debido al mantenimiento de los estados de conexión.
- Fallos en cascada: posibilidad de que los problemas de un servicio se propaguen rápidamente por el sistema.

## Comunicación asíncrona

Por el contrario, en la comunicación asíncrona, el cliente envía una solicitud a un servicio, pero no recibe una respuesta inmediata. En este caso, el cliente suele recibir solo un acuse de recibo de que la solicitud ha sido aceptada.

Los beneficios de la comunicación asíncrona incluyen:

- Soporte de arquitectura basada en eventos: ideal para los patrones de generación de eventos y de división de responsabilidades por consultas de comandos (CQRS).
- Mejor gestión de los recursos: capacidad de los servicios para procesar las solicitudes en función de su capacidad.
- Aislamiento de errores mejorado: disociación de los servicios, lo que evita los fallos en cascada.
- Gestión de los picos de carga: mejor gestión de los picos de tráfico mediante la puesta en cola de mensajes.

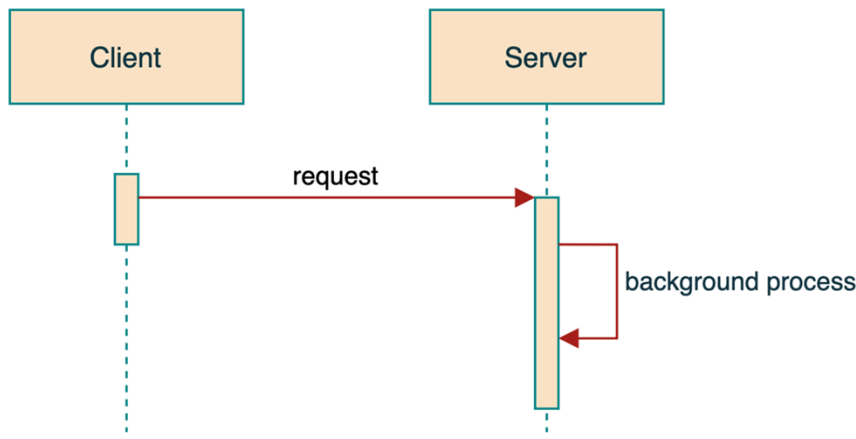
Entre los inconvenientes se incluye la complejidad. Por ejemplo:

- Si el cliente requiere el resultado de la operación asíncrona, implementar un mecanismo para obtener o recibir ese resultado requiere más esfuerzo.
- Puede resultar más difícil solucionar los problemas de las operaciones asíncronas, ya que la solución de problemas requiere examinar los registros de varios sistemas.
- Puede resultar más difícil probar las operaciones asíncronas, ya que las pruebas requieren la coordinación entre varios sistemas y servicios.

Entre los enfoques de la comunicación asíncrona se incluyen la comunicación “de disparo y olvido”, verificación de reclamaciones, devolución de llamadas y comunicación bidireccional.

## Disparo y olvido

En el patrón de disparo y olvido, un cliente envía una solicitud al servidor y recibe de forma sincronizada un acuse de recibo que indica que el servidor ha recibido el mensaje y lo procesará. Sin embargo, el procesamiento real aún no se ha realizado y el cliente no sabe cuándo o cómo se realizará. El siguiente diagrama ilustra este patrón.



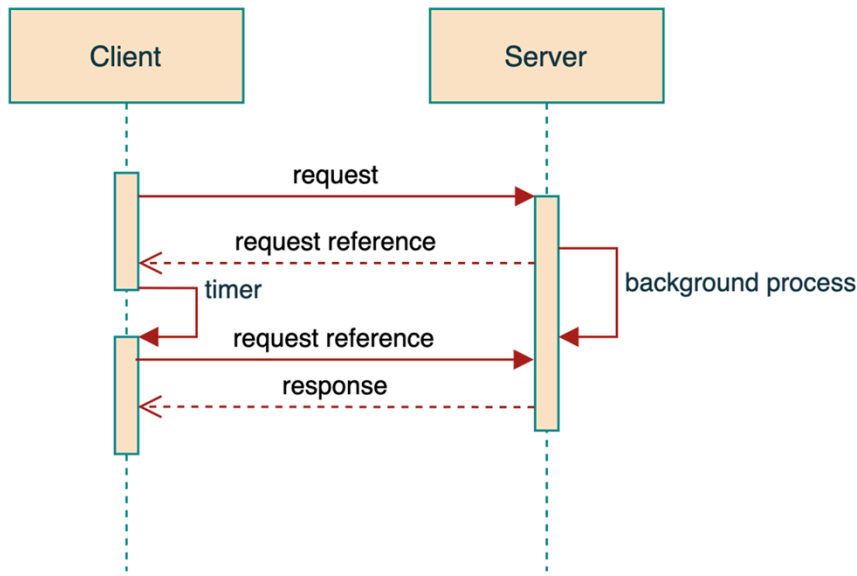
En este caso, el servicio no debería enviar el acuse de recibo hasta que el objeto haya persistido de forma duradera. Esta persistencia podría implementarse como una operación de escritura en una base de datos o al colocar un elemento en una cola.

Consideraciones adicionales:

- Implemente la idempotencia para gestionar los mensajes duplicados. Es decir, cada mensaje debe procesarse solo una vez.
- Tenga en cuenta las [colas de mensajes fallidos](#) en caso de errores de procesamiento.
- Monitoree las tasas de éxito del procesamiento de mensajes.

## Verificación de reclamación

Si un cliente necesita el resultado de una llamada de servicio, se puede crear el servicio para emitir una verificación de reclamación cuando reciba una solicitud. El siguiente diagrama ilustra este patrón. La verificación de reclamación se implementa como un identificador que el servicio devuelve en su acuse de recibo. El cliente puede usar este identificador más adelante para comprobar el estado de la solicitud y recuperar el resultado cuando se complete la solicitud.



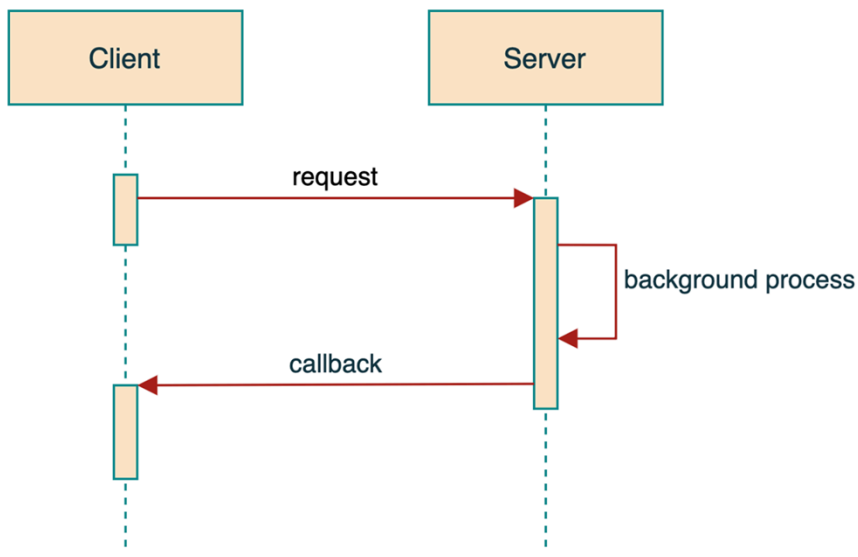
Los clientes deben implementar un mecanismo para sondear los resultados. Esto podría automatizarse (por ejemplo, se puede realizar una verificación cada n minutos) o implementarse de manera manual, en el caso de que la verificación se realice en respuesta a otro evento o acción del usuario. Los servicios que implementan el patrón de verificación de reclamaciones deben especificar explícitamente el período de validez de una verificación de reclamaciones.

Prácticas recomendadas:

- Implemente un retroceso exponencial en las votaciones.
- Establezca un tiempo de vida (time to live, TTL) adecuado para la verificación de las reclamaciones.
- Proporcione puntos de conexión de estado para el seguimiento del progreso.

## Devolución de llamada

En el patrón de devolución de llamadas, un cliente emite una solicitud a un servicio y proporciona una ubicación para que el servicio se ponga en contacto con él cuando se complete el procesamiento. El cliente no espera el resultado y el procesamiento continúa. El servicio es responsable de ponerse en contacto con la ubicación una vez finalizado el procesamiento y de proporcionar el resultado. Los tipos comunes de ubicaciones para las respuestas son REST APIs o colas. En el siguiente diagrama, se ilustra el patrón de devolución de llamada.

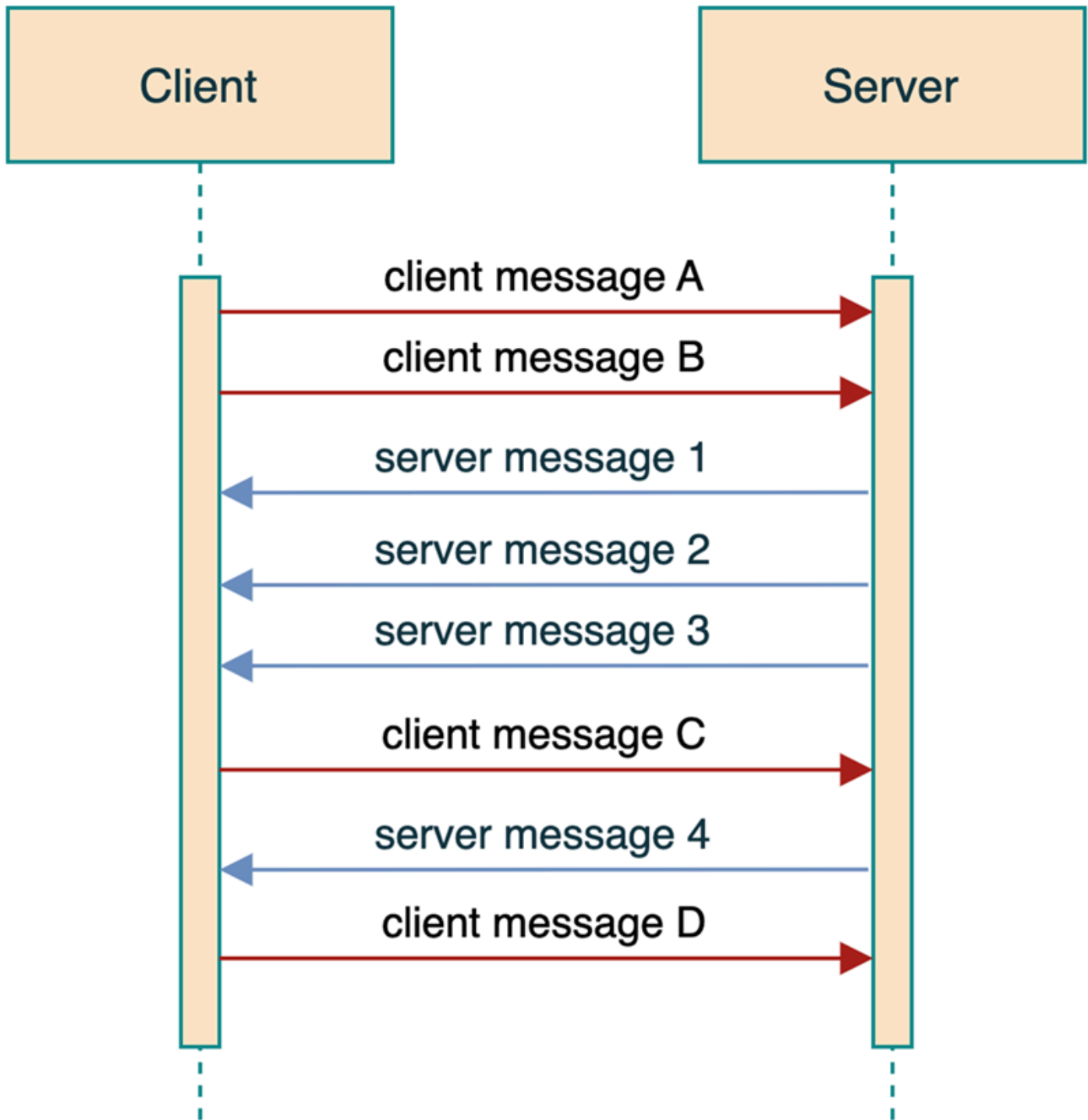


Implementación:

- Implemente mecanismos de reintento para las devoluciones de llamada fallidas.
- Proteja la ubicación de la devolución de la llamada como lo haría con otros servicios.
- Gestione los tiempos de espera de las devoluciones de llamadas.

## Comunicación bidireccional

Para implementar la comunicación bidireccional, debe crear una conexión con estado entre un cliente y un servicio, que permita que tanto el cliente como el servicio envíen y procesen mensajes. Esto se explica en el siguiente diagrama. Aunque la comunicación es asíncrona, el servicio debe poder admitir una conexión abierta para cada cliente.



Consideraciones de implementación:

- Ordenación de los mensajes
  - Números de secuencia

- Estrategias de partición
- Ordenación de los mensajes
- Administración de estados
  - Patrón de abastecimiento de eventos
  - Reconciliación de estado
  - Modelos de coherencia
- Gestión de errores
  - [Colas de mensajes fallidos](#)
  - Políticas de reintentos
  - [Interruptores](#)
  - Estrategias alternativas
- Supervisión y observabilidad
  - Correlación IDs
  - Seguimiento de mensajes
  - Métricas de desempeño
  - Indicadores de estado del sistema

# Opciones de coordinación

La comunicación síncrona y asíncrona funciona bien en el caso de un cliente que llama a un solo servicio o a un número reducido de servicios. Sin embargo, en un entorno real, esta comunicación puede complicarse rápido y ser difícil de escalar. La realización de una unidad de trabajo puede requerir varios microservicios, que a su vez pueden tener interdependencias. Con frecuencia, estas interacciones se modelan como un flujo de trabajo. Existen dos enfoques para el diseño de estos flujos de trabajo: la orquestación y la coreografía.

## Temas

- [Orquestación](#)
- [Coreografía](#)
- [Elección de su enfoque de coordinación](#)

## Orquestación

En este enfoque, un único orquestador es responsable de llamar a cada microservicio, de determinar si se deben realizar las llamadas en secuencia o en paralelo, de manipular las respuestas de los servicios individuales a lo largo del proceso y de compilar el resultado final. Un orquestador puede mezclar invocaciones asíncronas y síncronas.

[AWS Step Functions](#) y [Amazon Managed Workflows para Apache Airflow \(Amazon MWAA\)](#) son excelentes opciones para los coordinadores de flujos de trabajo.

La orquestación es una opción adecuada cuando hay ramas lógicas en el proceso y se necesita un único lugar donde encapsular esa lógica. También es útil cuando se quiere implementar el patrón asíncrono de verificación de reclamaciones. Por ejemplo, los flujos de trabajo estándar de Step Functions pueden poner en pausa un flujo de trabajo y esperar a que otro servicio le devuelva la llamada. El uso de un orquestador también mejora el monitoreo y la observabilidad de un proceso.

## Ejemplo: Step Functions

Puede usar Step Functions para coordinar varias funciones de Lambda y otras Servicios de AWS para crear flujos de trabajo complejos para la integración de microservicios. Esta opción resulta útil en especial en el caso de procesos prolongados y de varios pasos que implican varios microservicios.

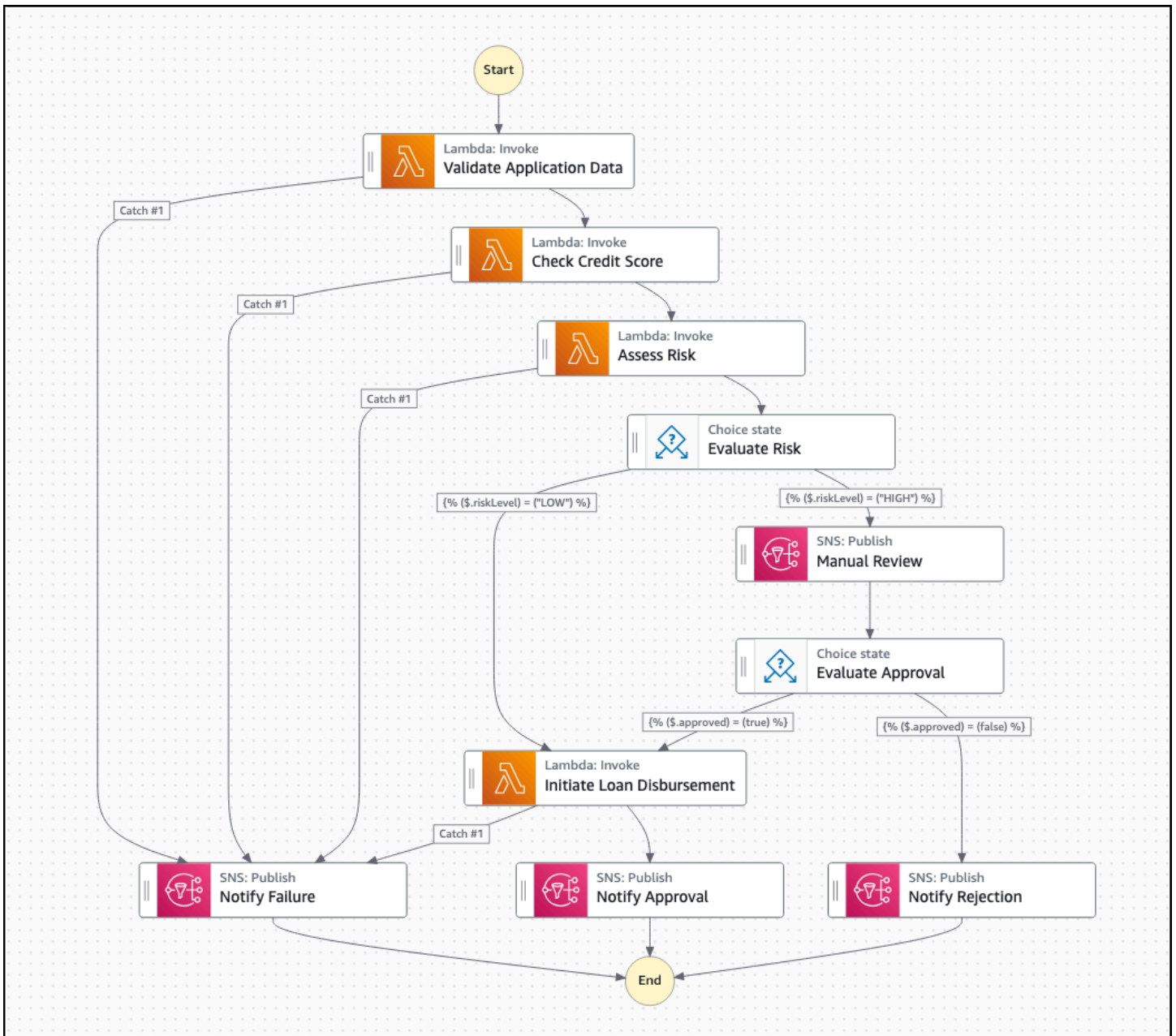
---

Debería considerar la posibilidad de utilizar Step Functions si:

- La integración de sus microservicios requiere de procesos complejos de varios pasos.
- Debe mantener el estado en todas las operaciones de larga duración.
- Desea implementar la lógica de gestión de errores y reintentos a nivel del flujo de trabajo.
- Debe coordinar operaciones síncronas y asíncronas.

Step Functions ofrece un editor visual para el diseño de flujos de trabajo complejos, lo que simplifica el proceso de creación y administración de máquinas de estados. Proporciona mecanismos integrados de gestión de errores, que incluyen la lógica de reintento y la administración del estado de los errores, lo que mejora la fiabilidad y la solidez de sus aplicaciones. Los flujos de trabajo estándar admiten procesos de larga duración de hasta un año, lo que es adecuado para flujos de trabajo que abarcan períodos prolongados. Esta opción separa la lógica de orquestación del código de la aplicación, por lo que reduce de manera importante la complejidad del código. De esta manera, los desarrolladores pueden centrarse en la lógica empresarial principal, mientras que Step Functions se encarga del control del flujo y la coordinación de los componentes distribuidos.

Por ejemplo, piense en el proceso de aprobación de un préstamo en una solicitud de servicios financieros, como se ilustra en el siguiente diagrama. El proceso comienza en el momento en que se presenta una solicitud de préstamo.



En la máquina de estados que se ilustra en el diagrama anterior, Step Functions orquesta los siguientes pasos:

- Validación de los datos de la aplicación (función de Lambda)
- Verificación de la calificación crediticia (función de Lambda que llama a una API externa)
- Evaluación del riesgo (función de Lambda)
- Si el riesgo es alto, procede a la revisión manual (tarea de aprobación humana)
- Si se aprueba, inicia el desembolso del préstamo (función de Lambda)

- Envío de notificación al solicitante (Amazon SNS)

Puede utilizar este enfoque para administrar de forma fiable un proceso complejo y potencialmente prolongado, con un sistema integrado de gestión de errores y la posibilidad de incluir pasos automatizados y manuales.

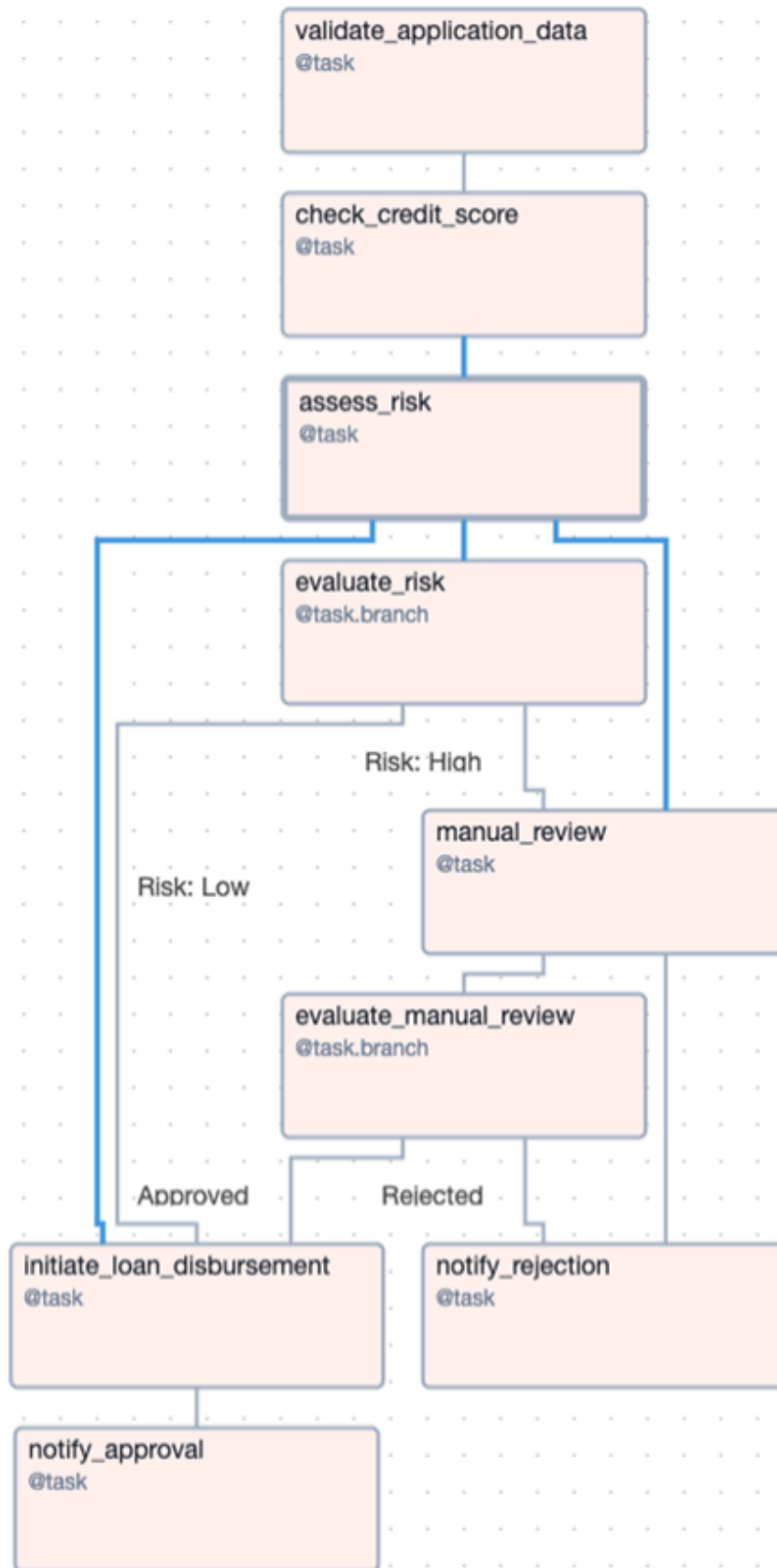
Consideraciones:

- Diseñe su máquina de estados con cuidado para gestionar todos los escenarios posibles.
- Realice los pasos en paralelo cuando sea posible.
- Utilice los mecanismos integrados de gestión de errores y reintento de Step Functions para los fallos permanentes y temporales.
- Piense en la posibilidad de utilizar [flujos de trabajo estándar o rápidos](#) según su caso de uso. Los flujos de trabajo rápidos pueden ser preferibles en el caso de los flujos de trabajo de corta duración o de gran volumen.
- [Monitoree las métricas de ejecución](#) para optimizar el flujo de trabajo.
- Utilice los flujos de trabajo anidados para encapsular y reutilizar la funcionalidad en varias máquinas de estado.
- En el caso de flujos de trabajo complejos, puede utilizar [Amazon Bedrock Agents](#) como alternativa a Step Functions.

Para obtener más información, consulte la [documentación de Step Functions](#).

## Ejemplo: Amazon MWAA

Si en su organización ya se utiliza Apache Airflow, Amazon MWAA es una opción natural como orquestador de flujos de trabajo. En Apache Airflow, los flujos de trabajo se crean como gráficos acíclicos dirigidos (DAGs) mediante Python. La representación en DAG de la máquina de estados que se ilustra en la sección Step Functions podría tener este aspecto:



Para obtener información sobre cómo trabajar con DAGs, consulte la documentación de [Amazon MWAA](#).

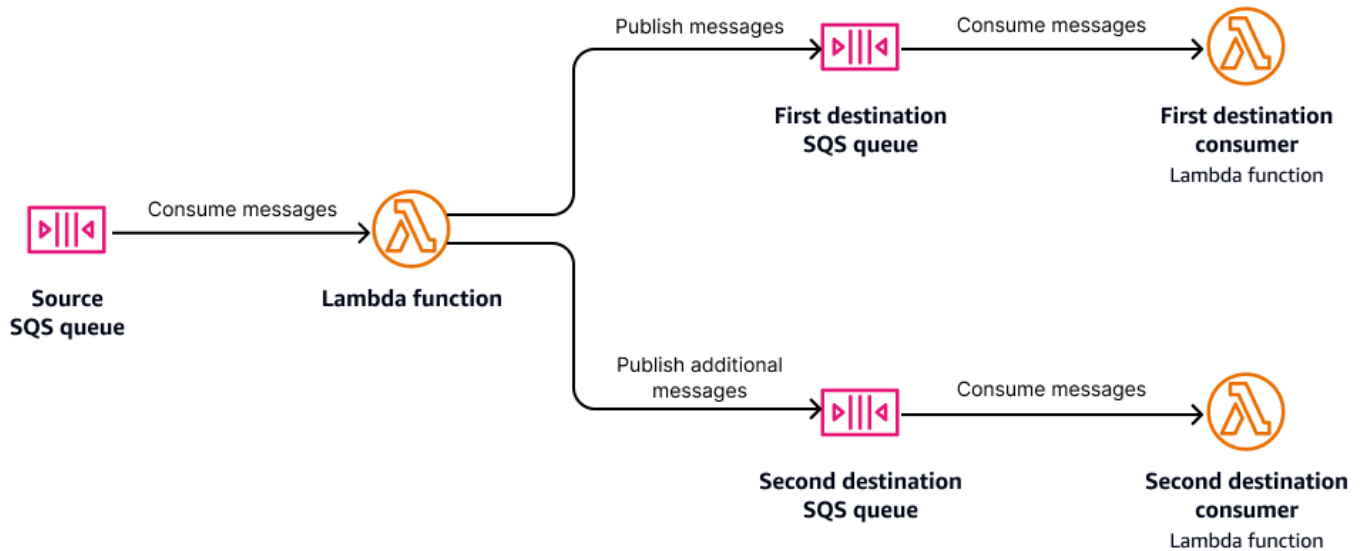
## Diferencias clave entre Step Functions y Amazon MWAA

- Step Functions es un servicio sin servidor totalmente administrado, lo que significa que no es necesario aprovisionar previamente ninguna infraestructura ni programar un período de mantenimiento. Amazon MWAA debe implementarse con anticipación y usted elige el tamaño y la cantidad de nodos de su clúster.
- En Step Functions, se pueden crear máquinas de estados de diversas formas, como Workflow Studio, directamente como JSON o mediante AWS Cloud Development Kit (AWS CDK). Apache Airflow DAGs está escrito en Python.
- Con Step Functions, no se incurrirá en costos si no hay flujos de trabajo en ejecución. Con Amazon MWAA, incurrirá en costes incluso cuando no DAGs esté en funcionamiento.

## Coreografía

En un sistema coreografiado, los componentes individuales reciben una tarea, realizan trabajos y, tal vez, emiten una tarea para realizar un trabajo posterior. No existe un mecanismo de orquestación central. La coreografía facilita el escalado de los servicios de manera independiente, ya que cada servicio funciona de manera bastante aislada. Realiza el trabajo cuando lo recibe, sea cual sea el rendimiento del servicio. La coreografía suele ser una parte central de una [arquitectura basada en eventos \(EDA\)](#).

En el siguiente diagrama, no existe coordinación entre las funciones de Lambda. Cada función procesa únicamente los mensajes de la cola suscrita. Cada función es responsable de su propia gestión de errores y puede controlar la simultaneidad. Es decir, si una dependencia descendente tiene un límite de solicitudes por segundo (RPS).



Una EDA ofrece una serie de ventajas, como el acoplamiento débil de los servicios y la extensibilidad. Esta guía no tiene como finalidad ofrecer un análisis completo de los principios de la EDA. Para obtener más información, consulte lo siguiente:

- [AWS Well-Architected Framework: lente de aplicaciones sin servidor](#)
- [Introduction to Event Driven Architecture](#) (Serverless Land)
- [Transitioning to event-driven architecture](#) (Serverless Developer Guide)

## Elección de su enfoque de coordinación

Tanto la coreografía como la orquestación tienen sus usos cuando se trata de integrar microservicios. Seleccione la coreografía dentro de los límites de un único microservicio, donde tendrá el control total sobre las dependencias. Seleccione la orquestación cuando trabaje más allá de los límites de los microservicios. Por ejemplo, varios microservicios que participan en una transacción distribuida se beneficiarán de la orquestación para dar cuenta de la reversión de los errores. Aquellos microservicios que gestionan eventos que podrían ser de interés para otros microservicios se beneficiarán de la coreografía y de una arquitectura basada en eventos.

Un patrón común para implementar la reversión cuando existen varios sistemas involucrados en una sola transacción es el patrón de la saga.

# Administrar APIs

Una gestión adecuada de las API hace que sus microservicios sean accesibles tanto para los consumidores internos como externos. AWS ofrece una variedad de servicios que puede usar juntos para exponer su APIs microservicio de forma segura. Estos servicios le permiten reforzar la seguridad e implementar la supervisión APIs y la observabilidad desde una ubicación central. También puedes usar [Amazon CloudFront](#) para mejorar el rendimiento si tus usuarios APIs se encuentran geográficamente alejados del lugar en el Región de AWS que están alojados los servicios.

## Amazon API Gateway

[Amazon API Gateway](#) es un servicio totalmente gestionado que permite a los desarrolladores crear, publicar, mantener, supervisar y proteger REST WebSocket APIs a cualquier escala. Se puede usar API Gateway para implementar muchos de los patrones descritos en la sección [Patrones de comunicación](#) de esta guía.

Existen dos tipos principales de REST APIs: REST y HTTP. Ambos tipos son compatibles RESTful APIs, pero ofrecen características diferentes. Para determinar cuál se adapta mejor a sus necesidades, consulte [Elegir entre REST APIs y HTTP APIs](#) en la documentación de API Gateway. Esta sección de la guía se centra en API Gateway REST APIs.

El uso de API Gateway como punto de entrada APIs proporciona un lugar único para implementar problemas comunes, como la validación y la seguridad de las solicitudes. API Gateway REST APIs ofrece [validación de solicitudes](#), que te permite definir el formato de tus solicitudes mediante el [esquema JSON](#). API Gateway valida las solicitudes entrantes según el esquema definido y rechaza las solicitudes con formato incorrecto.

## Autenticación y autorización

API Gateway REST APIs admite los siguientes mecanismos de autenticación (AuthN) y autorización (AuthZ):

- IAM: si se utiliza IAM, las solicitudes a la API deben firmarse mediante la [Signature Version 4 \(SigV4\) de AWS](#).
- Amazon Cognito: API Gateway validará el token portador presentado como emitido por un grupo de usuarios de Amazon Cognito. También se puede configurar un grupo de usuarios

de Amazon Cognito para que se integre con un proveedor de identidades (IdP) externo, si ya se está utilizando uno. También puede usar un grupo de usuarios de Amazon Cognito para la autenticación machine-to-machine (M2M).

- AWS Lambda autorizador: API Gateway invocará una función de Lambda que especifique para realizar las comprobaciones que desee a fin de determinar si una solicitud debe autorizarse.

Para obtener más información, consulte [Control y administración del acceso a REST APIs](#) en la documentación de API Gateway.

## Claves de API y límites de velocidad

Puedes controlar quién puede llamarte APIs y a qué velocidad mediante las claves de API y los planes de uso. Las claves de API no se deben usar para la autenticación, pero se pueden usar junto con los esquemas mencionados anteriormente. Los usuarios no siempre necesitan proporcionar su propia clave de API; por ejemplo, los autorizadores de Lambda pueden devolver una clave de API para un usuario. El plan de uso permite especificar el rendimiento, el límite de ráfaga y la cuota mensual. Para obtener más información, consulta [Planes de uso y claves de API para REST APIs](#) en la documentación de API Gateway.

## Público y privado APIs

Los REST de API Gateway a los APIs que se puede acceder a través de Internet admiten dos tipos de puntos finales:

- Optimizado para Edge, lo que significa que las solicitudes de las personas que llaman se envían a un CloudFront punto de presencia (POP) cercano. Esto puede mejorar el rendimiento de los clientes geográficamente dispares.
- Regional, lo que significa que las solicitudes se dirigen a un recurso dentro de una Región de AWS específica. Esta es una buena opción cuando todos los clientes están cerca de la región en la que está implementada su API.

API Gateway REST APIs también admite puntos de enlace de API privados, a los que se puede acceder desde una nube privada virtual (VPC) mediante un punto de enlace de VPC de interfaz. También puede compartir REST privado de forma segura APIs mediante la creación de puntos finales de VPC de interfaz en otros VPCs e incluso en otros. Cuentas de AWS Para obtener más información, consulta los [tipos de puntos de conexión de API para REST APIs](#) en la documentación de API Gateway.

## Cuándo usar API Gateway

API Gateway es una buena opción para servicios RESTful web y WebSocket conexiones en tiempo real. Cuando lo usas WebSocket APIs en API Gateway, puedes añadir comportamientos para los eventos de conexión y desconexión, como almacenar la conexión IDs en un banco de datos externo asociado a los atributos del cliente. También se pueden dirigir las solicitudes a comportamientos personalizados mediante los atributos de los mensajes.

Tanto REST como WebSocket APIs pueden integrarse directamente con muchos Servicios de AWS sin necesidad de recursos informáticos separados, como las funciones Lambda. Esto puede mejorar el rendimiento y reducir los costos.

REST APIs admite el enrutamiento basado en rutas y en encabezados, y puede usarlos por separado o juntos. Un patrón común es proporcionar una API REST como puerta de entrada para una serie de aplicaciones, con el fin de implementar problemas comunes APIs, tal como se ha explicado anteriormente, y luego comportarse como un proxy inverso y enrutar las solicitudes autorizadas al punto final de la API correcto.

# Mensajería

Como se explica en la sección [Patrones de comunicación](#), puede utilizar la mensajería para comunicarse de forma síncrona o asíncrona entre servicios. Hay muchos servicios de AWS sin servidor entre los que elegir, y su elección debe basarse en sus necesidades de integración. Por ejemplo, si se requiere una entrega de mensajes por pedido, debe utilizar un servicio de como Amazon Simple Queue Service (Amazon SQS) o Amazon Simple Notification Service (Amazon SNS). Ambos servicios admiten la entrega con el principio de primero en entrar, primero en salir (FIFO), a diferencia de Amazon EventBridge, que no lo hace.

En las siguientes secciones se tratan estos servicios de manera detallada.

## Temas

- [Amazon SQS](#)
- [Amazon SNS](#)
- [Amazon EventBridge](#)
- [AWS AppSync Events y API Gateway](#)

## Amazon SQS

[Amazon SQS](#) es compatible con colas estándar, que no garantizan el orden, y las colas FIFO, que sí garantizan el orden dentro de un grupo de mensajes determinado.

Las colas son un método común de coreografía de microservicios y proporcionan un almacenamiento duradero de los mensajes durante un máximo de 14 días. Las colas las llenan los productores y las vacían los consumidores. Cuando utiliza AWS Lambda como consumidor, puede configurar una cola de SQS como fuente de eventos. En este caso, la asignación de orígenes de eventos (ESM) del servicio Lambda sondea la cola por usted y envía los mensajes a su función de Lambda cuando están disponibles. Los microservicios que se ejecutan en otros tipos de servicios de cómputo, como Amazon Elastic Container Service (Amazon ECS) o Amazon Elastic Compute Cloud (Amazon EC2), deben implementar su propio mecanismo de sondeo para recuperar mensajes nuevos de la cola cuando estén disponibles.

Lambda ESM para Amazon SQS también admite el filtrado de mensajes, lo que permite procesar solo un subconjunto de mensajes de una cola en función del contenido del cuerpo del mensaje.

## Sondeo

Amazon SQS es compatible con sondeos cortos y largos de mensajes. El sondeo corto consulta un subconjunto de servidores para encontrar los mensajes disponibles y los devuelve de inmediato. Sin embargo, es posible que no se devuelvan todos los mensajes disponibles. Esto es útil cuando la aplicación necesita consumir los mensajes lo más rápido posible o no tolera la espera durante un tiempo prolongado.

Un sondeo prolongado aguarda hasta que haya transcurrido un período de tiempo configurable o se haya recibido un número configurable de mensajes antes de devolverlos. Esto podría reducir el número de sondeos vacíos, es decir, la cantidad de sondeos en los que no se devuelve ningún mensaje, en especial en el caso de las colas que no reciben muchos mensajes. Reducir el número de sondeos vacíos puede reducir los costos de Amazon SQS, ya que este servicio cobra por cada solicitud y cada operación de sondeo es una solicitud.

## Directrices

Las colas son una buena opción cuando:

- Desea desacoplar los componentes y no necesita una comunicación síncrona entre ellos.
- Se está comunicando entre componentes que tienen diferentes acuerdos de nivel de servicio (SLA) u objetivos de nivel de servicio (SLO) de disponibilidad diferentes.
- En general, tiene un único consumidor para un conjunto de mensajes.

Considere una opción alternativa si:

- Necesita una comunicación síncrona.
- Necesita una lógica de enrutamiento complicada para enviar los mensajes al consumidor correcto.

## Amazon SNS

[Amazon SNS](#) permite la creación de temas estándar y FIFO. Los temas se utilizan para implementar una arquitectura de publicación/suscripción (pub/sub). Amazon SNS admite varios tipos de suscripción, incluidos correo electrónico, SMS (siempre que haya configurado una identidad de origen, como un número gratuito o un código de 10 dígitos), puntos de conexión HTTP (S) y colas SQS. El suscriptor debe confirmar las suscripciones de los usuarios finales, como los mensajes

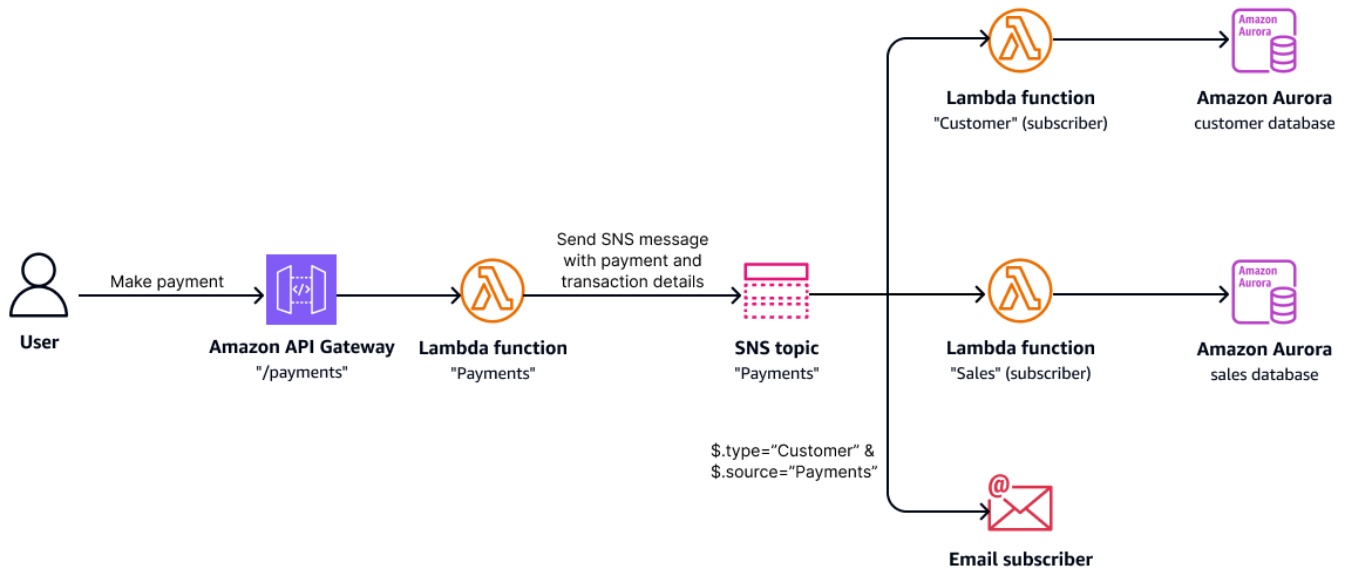
de correo electrónico y los mensajes SMS, a un tema de SNS. Amazon SNS permite que los servicios se distribuyan de manera amplia, lo que significa que se puede entregar un solo mensaje a un número potencialmente grande de suscriptores. Un tema estándar de SNS tiene un límite predeterminado de 12,5 millones de suscripciones.

En un entorno de microservicios, un tema de SNS resulta útil para desvincular la lógica de enrutamiento y entrega de mensajes de la lógica de un publicador. Es posible implementarlo mediante filtros de temas. De manera conceptual, los filtros de temas son ligeramente similares a las reglas de Amazon EventBridge, pero están configurados para cada suscriptor en lugar de estar disponibles desde una ubicación centralizada. Por ejemplo, supongamos que usted tiene:

- Un servicio de pedidos, que es el que procesa los pedidos.
- Un servicio de cumplimiento, encargado del cumplimiento de los pedidos.
- Un servicio de fidelización, que concede puntos de fidelidad a los miembros por sus pedidos.

Cuando un pedido está listo para ser procesado, publica un mensaje en un tema. El servicio de cumplimiento se suscribe al tema, pero no aplica filtros porque desea conocer todos los pedidos. Imagine que tiene un servicio de fidelización, que se encarga de conceder puntos a los miembros cuando hacen un pedido. Sin embargo, no todos los pedidos son realizados por los miembros. El servicio de fidelización se suscribiría al tema, pero implementaría un filtro de suscripción para verificar un atributo que indica si el pedido era para un miembro o para un invitado.

Piense en el caso en el que un sistema recibe una solicitud de un usuario final para realizar un pago, como se muestra en el siguiente diagrama. En este caso, varios sistemas intermedios necesitan saber que la solicitud se ha realizado para poder llevar a cabo diversas acciones. Cuando se utiliza Amazon SNS, los pagos se publican en un tema de SNS y las funciones de Lambda se suscriben al tema para actualizar las bases de datos tanto de clientes como de ventas. Además, una suscripción por correo electrónico (que el cliente debe confirmar) envía una confirmación (también por correo electrónico) al cliente mediante un filtro de suscripción.



## Directrices

Algunas de las funciones que se describen en esta sección para Amazon SNS se superponen con la funcionalidad que ofrece un bus de eventos, como EventBridge. Puede utilizar Amazon SNS cuando:

- Sabe que tendrá un gran número de suscriptores de un tema.
- Desea utilizar tipos de suscripción (como correo electrónico o mensajes SMS), que EventBridge no admite de forma nativa.
- Los suscriptores deberían tener la capacidad de determinar su filtro de suscripción.
- Necesita la entrega en orden a los suscriptores (por grupo de mensajes).

En el caso de que tenga muchos temas y se utilicen suscripciones y filtros para enrutar mensajes entre microservicios, es probable que EventBridge sea la mejor opción.

## Amazon EventBridge

[Amazon EventBridge](#) es un servicio de bus de eventos sin servidor que, en ocasiones, actúa como la base de una arquitectura basada en eventos (EDA). Es posible usarlo para enrutar y entregar mensajes de forma asíncrona entre microservicios. Con EventBridge, los productores publican eventos en un bus. Se pueden configurar reglas que coincidan con los eventos en función de su contenido, y seleccionar uno o más destinos a los que se enviarán los eventos que coincidan con

esa regla. EventBridge admite una [amplia gama de objetivos de reglas](#). El uso de un bus de eventos permite separar a los productores de los consumidores y consolidar la lógica de enrutamiento y entrega.

En EventBridge, también se pueden crear reglas programadas para realizar acciones en momentos específicos. Se pueden definir los eventos mediante expresiones basadas en cronómetros y tasas.

[EventBridge Pipes](#) permite canalizar mensajes de un [origen](#) a un [destino](#) sin necesidad de utilizar un servicio informático como AWS Lambda. Por ejemplo, supongamos que tiene una cola de SQS que recibe mensajes, lo que debería activar una máquina de estado AWS Step Functions. En lugar de crear una función de Lambda que tenga una asignación de orígenes de eventos para consumir los mensajes de la cola y escribir código mediante un AWS SDK para invocar la máquina de estado, puede usar EventBridge Pipes para hacerlo por usted sin necesidad de escribir ningún código personalizado.

EventBridge suele utilizarse con otros servicios de mensajería, como Amazon SQS y Amazon SNS. Por ejemplo, la entrega de eventos a una cola de SQS otorga al servicio receptor la flexibilidad de consumir mensajes cuando puede, a una velocidad independiente de la velocidad a la que se producen los eventos. Del mismo modo, se pueden organizar eventos que deberían estar dirigidos a un gran número de suscriptores de un tema de redes sociales.

## Directrices

Use EventBridge cuando:

- No necesita una comunicación síncrona entre los servicios.
- Desea desacoplar la lógica de enrutamiento de mensajes de los microservicios. Los microservicios solo producen eventos y los publican en el bus de eventos, y los servicios interesados crean las reglas para hacer coincidir esos eventos y publicarlos.
- Se deben enviar los mensajes de un servicio compatible a otro.

Considere la posibilidad de utilizar otros servicios en situaciones en las que:

- Se necesita un orden estricto de los eventos. En estos casos, se pueden utilizar colas FIFO de Amazon SQS o temas FIFO de Amazon SNS. También se pueden utilizar utilizar servicios de transmisión de eventos de como Amazon Kinesis Data Streams o Amazon Managed Streaming para Apache Kafka (Amazon MSK).

# AWS AppSync Events y API Gateway

Tanto AWS AppSync Events como Amazon API Gateway proporcionan una experiencia de WebSocket administrada para sus microservicios.

[AWS AppSync Events](#) ofrece una experiencia simplificada para la mensajería en tiempo real mediante el uso de un WebSocket. AWS AppSync Events admite la mensajería de unidifusión y multidifusión así como la agrupación flexible de canales en espacios de nombres, con compatibilidad con caracteres comodín. Los microservicios se pueden comunicar entre sí de diversas formas mediante AWS AppSync Events. Por ejemplo, un servicio que recibe datos en tiempo real puede transformarlos y publicarlos en el canal correcto, donde los suscriptores los recibirán en tiempo real.

[API Gateway](#) también es compatible con las API de WebSocket. Puede definir integraciones con Servicios de AWS, por ejemplo, Amazon DynamoDB y AWS Lambda, y configurar las expresiones de selección de rutas que se asignen a esas integraciones. API Gateway tiene rutas especiales que se pueden usar para autorizar y administrar tus conexiones de WebSocket. Según sus necesidades, se puede almacenar la información de conexión de WebSocket en un almacén de datos como DynamoDB. Con esta información, los mensajes se pueden publicar en conexiones WebSocket específicas a través de una API de REST, con un ID de conexión específico.

## Directrices

Use AWS AppSync Events cuando:

- Tiene varios canales de mensajería agrupados en espacios de nombres y desea publicar grupos de canales y suscribirse a ellos mediante caracteres comodín.
- Sus comunicaciones se realizan principalmente entre diferentes sistemas y no entre Servicios de AWS.

Use API de WebSocket de API Gateway cuando:

- Desea permitir que los clientes tengan conexiones persistentes a las integraciones del Servicio de AWS en tiempo real.
- Desea administrar usted las conexiones de WebSocket. Por ejemplo, tal vez desee permitir que otros sistemas envíen mensajes a un cliente en particular después de buscar su ID de conexión.
- Desea utilizar las características de API Gateway, como las implementaciones por etapas o las integraciones de proxy, o desea configurar sus propios subprotocolos.

## Preguntas frecuentes

### ¿Cómo puedo combinar diferentes patrones de integración?

En la mayoría de las situaciones, querrá combinar patrones de integración. Por ejemplo, puede utilizar AWS Step Functions para orquestar un proceso que llame a un servicio remoto mediante el patrón de verificación de reclamaciones. O bien, se puede tener un proceso orquestado que coloque los mensajes en colas, lo que, a su vez, desencadena servicios coreografiados.

### ¿Cuál es la principal ventaja de utilizar una arquitectura de microservicios?

Las principales ventajas incluyen el escalado independiente de los servicios, la mejora del aislamiento de errores, la mejora de la velocidad de desarrollo mediante el trabajo en equipo en paralelo y la capacidad de entrega e implementación continuas (CI/CD).

### ¿Cómo puedo implementar la gestión de errores en estos patrones?

Puede implementar la gestión de errores mediante los mecanismos integrados en Servicios de AWS. Por ejemplo, AWS Lambda las funciones se pueden configurar con una lógica de reintentos, y Amazon SQS admite colas de letra muerta para gestionar los errores persistentes. Además, Step Functions proporciona mecanismos de control y reintento de errores a nivel del flujo de trabajo.

### ¿Cuáles son los beneficios de utilizar el patrón de verificación de reclamaciones en la comunicación asíncrona?

El patrón de verificación de reclamaciones permite a los clientes recibir un identificador al enviar la solicitud. Este identificador se puede utilizar más adelante para comprobar el estado y recuperar el resultado. Este patrón beneficia a los clientes al proporcionar un mecanismo para sondear los resultados sin esperar de forma síncrona. Para obtener más información, consulte la sección [Verificación de notificación](#) mencionada previamente en esta guía.

## ¿Cómo mejora el patrón de devolución de llamada la comunicación asíncrona en los microservicios?

El patrón de devolución de llamadas mejora la comunicación asíncrona al permitir que el cliente proporcione una ubicación para que el servicio se contacte una vez finalizado el procesamiento. Esto evita que el cliente espere una respuesta y permite continuar con otras tareas. Para obtener más información, consulte la sección [Devolución de llamadas](#) mencionada previamente en esta guía.

## ¿Puedo implementar la comunicación bidireccional en los microservicios mediante los patrones descritos?

Puede implementar la comunicación bidireccional al crear una conexión con estado entre un cliente y un servicio, de modo que ambos puedan enviar y procesar mensajes de forma asíncrona. Esto requiere que el servicio admita una conexión abierta para cada cliente. Para obtener más información, consulte la sección [Comunicación bidireccional](#) mencionada previamente en esta guía.

## ¿Cómo puedo optimizar el uso de las funciones de Lambda en los patrones de comunicación asíncronos?

Puede optimizar las funciones de Lambda al asegurarse de que son idempotentes para gestionar posibles duplicaciones de mensajes, mediante el uso de las características de Amazon SQS, como los grupos de mensajes para realizar pedidos, y la implementación de sondeos prolongados para reducir los costos. Además, puede monitorear las métricas de ejecución para identificar oportunidades de optimización.

## ¿Cuáles son las principales diferencias entre el uso de Amazon SNS y EventBridge el patrón? pub/sub

Amazon SNS envía un único mensaje a todos los suscriptores, que puede incluir datos innecesarios para algunos suscriptores. Amazon EventBridge permite un control más detallado al permitirle tener varias reglas que coincidan con un solo evento, y cada regla desencadena un servicio o acción posterior diferente. Para obtener más información, consulte [Amazon SNS](#) y [EventBridge](#) las secciones anteriores de esta guía.

# Recursos

## Servicio de AWS Documentación de

- [Amazon API Gateway](#)
- [AWS AppSync Eventos de](#)
- [Amazon EventBridge](#)
- [Amazon MWAA](#)
- [Amazon SNS](#)
- [Amazon SQS](#)
- [AWS Step Functions](#)

## Lecturas adicionales

- [Estrategia para modernizar las aplicaciones en la nube de Nube de AWS](#)
- [Enfoque gradual para modernizar las aplicaciones en la nube de Nube de AWS](#)
- [Evaluación de la preparación para la modernización de las aplicaciones en la nube de Nube de AWS](#)
- [Descomposición de monolitos en microservicios](#)
- [Implementación de patrones de integración empresarial con servicios de AWS de mensajería: canales punto a punto](#)
- [Mensajería de pub/sub: notificaciones de eventos asíncronas](#)

## Historial de documentos

En la siguiente tabla, se describen cambios significativos de esta guía. Si quiere recibir notificaciones de futuras actualizaciones, puede suscribirse a las [notificaciones RSS](#).

Cambio	Descripción	Fecha
<a href="#">Actualizaciones importantes</a>	Guía ampliada, reorganizada y actualizada para reflejar las actualizaciones de los Servicio de AWS.	10 de septiembre de 2025
<a href="#">Publicación inicial</a>	—	11 de enero de 2021

# AWS Glosario de orientación prescriptiva

Los siguientes son términos de uso común en las estrategias, guías y patrones proporcionados por la Guía AWS prescriptiva. Para sugerir entradas, utilice el enlace [Enviar comentarios](#) al final del glosario.

## Números

### Las 7 R

Siete estrategias de migración comunes para trasladar aplicaciones a la nube. Estas estrategias se basan en las 5 R que Gartner identificó en 2011 y consisten en lo siguiente:

- **Refactor/re-architect** — Mueva una aplicación y modifique su arquitectura aprovechando al máximo las funciones nativas de la nube para mejorar la agilidad, el rendimiento y la escalabilidad. Por lo general, esto implica trasladar el sistema operativo y la base de datos. Ejemplo: migre su base de datos Oracle local a la PostgreSQL-Compatible edición Amazon Aurora.
- **Redefinir la plataforma (transportar y redefinir)**: traslade una aplicación a la nube e introduzca algún nivel de optimización para aprovechar las capacidades de la nube. Ejemplo: Migrar la base de datos Oracle en las instalaciones a Amazon Relational Database Service (Amazon RDS) para Oracle en la nube de Nube de AWS.
- **Recomprar (readquirir)**: cambie a un producto diferente, lo cual se suele llevar a cabo al pasar de una licencia tradicional a un modelo SaaS. Ejemplo: migre su sistema de gestión de relaciones con los clientes (CRM) a Salesforce.com.
- **Volver a alojar (migrar mediante lift-and-shift)**: traslade una aplicación a la nube sin hacer cambios para aprovechar las funcionalidades de la nube. Ejemplo: Migrar la base de datos de Oracle en las instalaciones a Oracle en una instancia de EC2 en la Nube de AWS.
- **Reubicar**: (migrar el hipervisor mediante lift and shift): traslade la infraestructura a la nube sin comprar equipo nuevo, reescribir aplicaciones o modificar las operaciones actuales. Los servidores se migran de una plataforma en las instalaciones a un servicio en la nube para la misma plataforma. Ejemplo: migrar una Microsoft Hyper-V aplicación a AWS.
- **Retener (revisitar)**: conserve las aplicaciones en el entorno de origen. Estas pueden incluir las aplicaciones que requieren una refactorización importante, que desee posponer para más adelante, y las aplicaciones heredadas que desee retener, ya que no hay ninguna justificación empresarial para migrarlas.

- Retirar: retire o elimine las aplicaciones que ya no sean necesarias en un entorno de origen.

## A

### A2A () Agent-to-Agent

Un protocolo completo para la colaboración entre agentes que facilita la delegación de tareas y la transferencia de estados.

### ABAC

Consulte [control de acceso basado en atributos](#).

### servicios abstractos

Consulte [servicios administrados](#).

### ACID

Consulte [atomicidad, consistencia, aislamiento, durabilidad](#).

### migración activa-activa

Método de migración de bases de datos en el que las bases de datos de origen y destino se mantienen sincronizadas (mediante una herramienta de replicación bidireccional o mediante operaciones de escritura doble) y ambas bases de datos gestionan las transacciones de las aplicaciones conectadas durante la migración. Este método permite la migración en lotes pequeños y controlados, en lugar de requerir una transición única. Es más flexible, pero requiere más trabajo que una [migración activa-pasiva](#).

### migración activa-pasiva

Método de migración de bases de datos en el que las bases de datos de origen y destino se mantienen sincronizadas, pero solo la de origen gestiona las transacciones de las aplicaciones conectadas, mientras los datos se replican en la de destino. La base de datos de destino no acepta ninguna transacción durante la migración.

### Agente

Un sistema de IA que puede razonar, planificar y tomar medidas de forma autónoma utilizando herramientas para alcanzar los objetivos.

## Agent Ops

Prácticas operativas para crear, probar, implementar y ejecutar agentes de IA en producción a escala.

### función de agregación

Función SQL que actúa en un grupo de filas y calcula un único valor de devolución para el grupo. Entre los ejemplos de funciones de agregación se incluyen SUM y MAX.

## IA

Consulte [inteligencia artificial](#).

## AIOps

Consulte [operaciones de inteligencia artificial](#)

### anonimización

El proceso de eliminar permanentemente la información personal de un conjunto de datos. La anonimización puede ayudar a proteger la privacidad personal. Los datos anonimizados ya no se consideran datos personales.

### antipatronos

Una solución que se utiliza con frecuencia para un problema recurrente en el que la solución es contraproducente, ineficaz o menos eficaz que una alternativa.

### control de aplicaciones

Enfoque de seguridad que permite usar de manera exclusiva aplicaciones aprobadas para ayudar a proteger un sistema contra el malware.

### cartera de aplicaciones

Recopilación de información detallada sobre cada aplicación que utiliza una organización, incluido el costo de creación y mantenimiento de la aplicación y su valor empresarial. Esta información es clave para [el proceso de detección y análisis de la cartera](#) y ayuda a identificar y priorizar las aplicaciones que se van a migrar, modernizar y optimizar.

### inteligencia artificial (IA)

El campo de la informática que se dedica al uso de tecnologías informáticas para realizar funciones cognitivas que suelen estar asociadas a los seres humanos, como el aprendizaje, la resolución de problemas y el reconocimiento de patrones. Para más información, consulte [¿Qué es la inteligencia artificial?](#)

## operaciones de inteligencia artificial (AIOps)

El proceso de utilizar técnicas de machine learning para resolver problemas operativos, reducir los incidentes operativos y la intervención humana, y mejorar la calidad del servicio. Para obtener más información sobre cómo se utiliza AIOps en la estrategia de migración de AWS, consulte la [Guía de integración de operaciones](#).

## cifrado asimétrico

Algoritmo de cifrado que utiliza un par de claves, una clave pública para el cifrado y una clave privada para el descifrado. Puede compartir la clave pública porque no se utiliza para el descifrado, pero el acceso a la clave privada debe estar sumamente restringido.

## atomicidad, consistencia, aislamiento, durabilidad (ACID)

Conjunto de propiedades de software que garantizan la validez de los datos y la fiabilidad operativa de una base de datos, incluso en caso de errores, cortes de energía u otros problemas.

## control de acceso basado en atributos (ABAC)

La práctica de crear permisos detallados basados en los atributos del usuario, como el departamento, el puesto de trabajo y el nombre del equipo. Para obtener más información, consulte [ABAC AWS en la](#) documentación AWS Identity and Access Management (IAM).

## origen de datos fidedigno

Ubicación en la que se almacena la versión principal de los datos, que se considera la fuente de información más fiable. Puede copiar los datos del origen de datos autorizado a otras ubicaciones con el fin de procesarlos o modificarlos, por ejemplo, anonimizarlos, redactarlos o seudonimizarlos.

## Zona de disponibilidad

Una ubicación distinta dentro de una Región de AWS que está aislada de los fallos en otras zonas de disponibilidad y que proporciona una conectividad de red económica y de baja latencia a otras zonas de disponibilidad de la misma región.

## AWS Marco de adopción de la nube (AWS CAF)

Un marco de directrices y mejores prácticas AWS para ayudar a las organizaciones a desarrollar un plan eficiente y eficaz para migrar con éxito a la nube. AWS CAF organiza la orientación en seis áreas de enfoque denominadas perspectivas: negocios, personas, gobierno, plataforma, seguridad y operaciones. Las perspectivas empresariales, humanas y de gobernanza se centran en las habilidades y los procesos empresariales; las perspectivas de plataforma, seguridad y

operaciones se centran en las habilidades y los procesos técnicos. Por ejemplo, la perspectiva humana se dirige a las partes interesadas que se ocupan de los Recursos Humanos (RR. HH.), las funciones del personal y la administración de las personas. Desde esta perspectiva, AWS CAF proporciona orientación para el desarrollo, la formación y la comunicación de las personas a fin de preparar a la organización para una adopción exitosa de la nube. Para obtener más información, consulte la [Página web de AWS CAF](#) y el [Documento técnico de AWS CAF](#).

#### AWS Marco de calificación de la carga de trabajo (AWS WQF)

Herramienta que evalúa las cargas de trabajo de migración de bases de datos, recomienda estrategias de migración y proporciona estimaciones de trabajo. AWS WQF se incluye con AWS Schema Conversion Tool (). AWS SCT Analiza los esquemas de bases de datos y los objetos de código, el código de las aplicaciones, las dependencias y las características de rendimiento y proporciona informes de evaluación.

## B

#### bot malicioso

[Bot](#) destinado a causar interrupciones o daños a personas u organizaciones.

#### BCP

Consulte [planificación de la continuidad del negocio](#).

#### gráfico de comportamiento

Una vista unificada e interactiva del comportamiento de los recursos y de las interacciones a lo largo del tiempo. Puede utilizar un gráfico de comportamiento con Amazon Detective para examinar los intentos de inicio de sesión fallidos, las llamadas sospechosas a la API y acciones similares. Para obtener más información, consulte [Datos en un gráfico de comportamiento](#) en la documentación de Detective.

#### sistema big-endian

Un sistema que almacena primero el byte más significativo. Consulte también [endianidad](#).

#### clasificación binaria

Un proceso que predice un resultado binario (una de las dos clases posibles). Por ejemplo, es posible que su modelo de ML necesite predecir problemas como “¿Este correo electrónico es spam o no es spam?” o “¿Este producto es un libro o un automóvil?”.

## filtro de floración

Estructura de datos probabilística y eficiente en términos de memoria que se utiliza para comprobar si un elemento es miembro de un conjunto.

## blue/green despliegue

Estrategia de implementación en la que se crean dos entornos separados, pero idénticos. La versión actual de la aplicación se ejecuta en un entorno (azul) y la nueva versión de la aplicación se ejecuta en el otro entorno (verde). Esta estrategia lo ayuda a hacer reversiones rápidas con un impacto mínimo.

## bot

Aplicación de software que ejecuta tareas automatizadas a través de Internet y simula la actividad o interacción humana. Algunos bots son útiles o beneficiosos, como los rastreadores web que indexan la información de Internet. Otros bots, conocidos como bots maliciosos, tienen como objetivo causar interrupciones o daños a personas u organizaciones.

## botnet

Redes de [bots](#) infectadas por [malware](#) y que están bajo el control de una sola parte, conocida como pastor de bots u operador de bots. Las botnets son el mecanismo más conocido para escalar los bots y su impacto.

## branch

Área contenida de un repositorio de código. La primera rama que se crea en un repositorio es la rama principal. Puede crear una rama nueva a partir de una rama existente y, a continuación, desarrollar características o corregir errores en la rama nueva. Una rama que se genera para crear una característica se denomina comúnmente rama de característica. Cuando la característica se encuentra lista para su lanzamiento, se vuelve a combinar la rama de característica con la rama principal. Para obtener más información, consulte [Acerca de las sucursales](#) (GitHub documentación).

## acceso de emergencia

En circunstancias excepcionales y mediante un proceso aprobado, es una forma rápida de que un usuario pueda acceder a un Cuenta de AWS sitio al que normalmente no tiene permisos de acceso. Para obtener más información, consulte el indicador de [implementación de procedimientos rompe-cristales](#) en la AWS Well-Architected guía.

## estrategia de implementación sobre infraestructura existente

La infraestructura existente en su entorno. Al adoptar una estrategia de implementación sobre infraestructura existente para una arquitectura de sistemas, se diseña la arquitectura en función de las limitaciones de los sistemas y la infraestructura actuales. Si está ampliando la infraestructura existente, puede combinar las estrategias de implementación sobre infraestructuras existentes y de [implementación desde cero](#).

## caché de búfer

El área de memoria donde se almacenan los datos a los que se accede con más frecuencia.

## capacidad empresarial

Lo que hace una empresa para generar valor (por ejemplo, ventas, servicio al cliente o marketing). Las arquitecturas de microservicios y las decisiones de desarrollo pueden estar impulsadas por las capacidades empresariales. Para obtener más información, consulte la sección [Organizado en torno a las capacidades empresariales](#) del documento técnico [Ejecutar microservicios en contenedores en AWS](#).

## planificación de la continuidad del negocio (BCP)

Plan que aborda el posible impacto de un evento disruptivo, como una migración a gran escala en las operaciones y permite a la empresa reanudar las operaciones rápidamente.

# C

## CAF

Consulte [AWS Cloud Adoption Framework](#).

## implementación canario

Lanzamiento lento e incremental de una versión para los usuarios finales. Cuando tenga mayor confianza en la nueva versión, la implementa y reemplaza la versión actual en su totalidad.

## CCoE

Consulte [Centro de excelencia en la nube](#).

## CDC

Consulte [captura de datos de cambios](#).

## captura de datos de cambio (CDC)

Proceso de seguimiento de los cambios en un origen de datos, como una tabla de base de datos, y registro de los metadatos relacionados con el cambio. Puede utilizar los CDC para diversos fines, como auditar o replicar los cambios en un sistema de destino para mantener la sincronización.

## ingeniería del caos

Introducción intencionada de fallos o eventos disruptivos para poner a prueba la resiliencia de un sistema. Puedes usar [AWS Fault Injection Service \(AWS FIS\)](#) para realizar experimentos que estresen tus AWS cargas de trabajo y evalúen su respuesta.

## CI/CD

Consulte [integración continua y entrega continua](#).

## clasificación

Un proceso de categorización que permite generar predicciones. Los modelos de ML para problemas de clasificación predicen un valor discreto. Los valores discretos siempre son distintos entre sí. Por ejemplo, es posible que un modelo necesite evaluar si hay o no un automóvil en una imagen.

## Desarrollador ciudadano

Un usuario empresarial que crea aplicaciones de IA utilizando plataformas sin code/low código sin conocimientos técnicos especializados.

## cifrado del cliente

Cifrado de datos localmente, antes de que el objetivo los Servicio de AWS reciba.

## Centro de excelencia en la nube (CCoE)

Equipo multidisciplinario que impulsa los esfuerzos de adopción de la nube en toda la organización, incluido el desarrollo de las prácticas recomendadas en la nube, la movilización de recursos, el establecimiento de plazos de migración y la dirección de la organización durante las transformaciones a gran escala. Para obtener más información, consulte las [publicaciones de CCoE](#) en el blog de estrategia Nube de AWS empresarial.

## computación en la nube

La tecnología en la nube que se utiliza normalmente para la administración de dispositivos de IoT y el almacenamiento de datos de forma remota. La computación en la nube suele estar relacionada con la tecnología de [computación de periferia](#).

## modelo operativo en la nube

En una organización de TI, el modelo operativo que se utiliza para crear, madurar y optimizar uno o más entornos de nube. Para obtener más información, consulte [Creación de su modelo operativo de nube](#).

## etapas de adopción de la nube

Las siguientes son las cuatro fases por las que suelen pasar las empresas cuando migran a la Nube de AWS:

- Proyecto: ejecución de algunos proyectos relacionados con la nube con fines de prueba de concepto y aprendizaje
- Fundamento: realización de inversiones fundamentales para escalar la adopción de la nube (p. ej., crear una zona de aterrizaje, definir un CCoE, establecer un modelo de operaciones)
- Migración: migración de aplicaciones individuales
- Re-invention — Optimizar los productos y servicios e innovar en la nube

Stephen Orban definió estas etapas en la entrada del blog The [Journey Toward Cloud-First & the Stages of Adoption del](#) blog Nube de AWS Enterprise Strategy. Para obtener información sobre su relación con la estrategia de AWS migración, consulte la [guía de preparación para la migración](#).

## CMDB

Consulte [base de datos de administración de configuración](#).

## repositorio de código

Una ubicación donde el código fuente y otros activos, como documentación, muestras y scripts, se almacenan y actualizan mediante procesos de control de versiones. Algunos repositorios en la nube comunes son GitHub o Bitbucket Cloud. Cada versión del código se denomina rama. En una estructura de microservicios, cada repositorio se encuentra dedicado a una única funcionalidad. Una sola CI/CD canalización puede utilizar varios repositorios.

## caché en frío

Una caché de búfer que está vacía no está bien poblada o contiene datos obsoletos o irrelevantes. Esto afecta al rendimiento, ya que la instancia de la base de datos debe leer desde la memoria principal o el disco, lo que es más lento que leer desde la memoria caché del búfer.

## datos fríos

Datos a los que se accede con poca frecuencia y que suelen ser históricos. Al consultar este tipo de datos, normalmente se aceptan consultas lentas. Trasladar estos datos a niveles o clases de almacenamiento de menor rendimiento y menos costosos puede reducir los costos.

## visión artificial (CV)

Campo de la [IA](#) que utiliza el machine learning para analizar y extraer información de formatos visuales, como imágenes y videos digitales. Por ejemplo, Amazon SageMaker AI proporciona algoritmos de procesamiento de imágenes para CV.

## deriva de configuración

En el caso de una carga de trabajo, un cambio en la configuración con respecto al estado esperado. Podría provocar que la carga de trabajo deje de cumplir las normas y, por lo general, es gradual e involuntaria.

## base de datos de administración de configuración (CMDB)

Repositorio que almacena y administra información sobre una base de datos y su entorno de TI, incluidos los componentes de hardware y software y sus configuraciones. Por lo general, los datos de una CMDB se utilizan en la etapa de detección y análisis de la cartera de productos durante la migración.

## paquete de conformidad

Un conjunto de AWS Config reglas y medidas correctivas que puede reunir para personalizar sus controles de conformidad y seguridad. Puede implementar un paquete de conformidad como una entidad única en una región Cuenta de AWS y, o en una organización, mediante una plantilla YAML. Para obtener más información, consulta los [paquetes de conformidad](#) en la documentación. AWS Config

## integración y entrega continuas (I) CI/CD

El proceso de automatización de las etapas de origen, creación, prueba, puesta en escena y producción del proceso de publicación del software. CI/CD se describe comúnmente como una canalización. CI/CD puede ayudarlo a automatizar los procesos, mejorar la productividad, mejorar

la calidad del código y entregar más rápido. Para obtener más información, consulte [Beneficios de la entrega continua](#). CD también puede significar implementación continua. Para obtener más información, consulte [Entrega continua frente a implementación continua](#).

## CV

Consulte [visión artificial](#).

## D

### datos en reposo

Datos que están estacionarios en la red, como los datos que se encuentran almacenados.

### clasificación de datos

Un proceso para identificar y clasificar los datos de su red en función de su importancia y sensibilidad. Es un componente fundamental de cualquier estrategia de administración de riesgos de ciberseguridad porque lo ayuda a determinar los controles de protección y retención adecuados para los datos. La clasificación de los datos es un componente del pilar de seguridad del AWS Well-Architected Framework. Para obtener más información, consulte [Clasificación de datos](#).

### deriva de datos

Una variación significativa entre los datos de producción y los datos que se utilizaron para entrenar un modelo de machine learning, o un cambio significativo en los datos de entrada a lo largo del tiempo. La deriva de datos puede reducir la calidad, la precisión y la imparcialidad generales de las predicciones de los modelos de machine learning.

### datos en tránsito

Datos que se mueven de forma activa por la red, por ejemplo, entre los recursos de la red.

### mallado de datos

Marco de arquitectura que proporciona una propiedad de datos distribuida y descentralizada con una administración y una gobernanza centralizadas.

### minimización de datos

El principio de recopilar y procesar solo los datos estrictamente necesarios. Practicar la minimización de los datos Nube de AWS puede reducir los riesgos de privacidad, los costos y la huella de carbono de la analítica.

## perímetro de datos

Un conjunto de barreras preventivas en su AWS entorno que ayudan a garantizar que solo las identidades confiables accedan a los recursos confiables desde las redes esperadas. Para obtener más información, consulte [Crear un perímetro de datos sobre](#). AWS

## preprocesamiento de datos

Transformar los datos sin procesar en un formato que su modelo de ML pueda analizar fácilmente. El preprocesamiento de datos puede implicar eliminar determinadas columnas o filas y corregir los valores faltantes, incoherentes o duplicados.

## procedencia de los datos

El proceso de rastrear el origen y el historial de los datos a lo largo de su ciclo de vida, por ejemplo, la forma en que se generaron, transmitieron y almacenaron los datos.

## titular de los datos

Persona cuyos datos se recopilan y procesan.

## almacenamiento de datos

Sistema de administración de datos que respalda la inteligencia empresarial, como los análisis. Los almacenes de datos suelen contener grandes cantidades de datos históricos y, por lo general, se utilizan para las consultas y los análisis.

## lenguaje de definición de datos (DDL)

Instrucciones o comandos para crear o modificar la estructura de tablas y objetos de una base de datos.

## lenguaje de manipulación de datos (DML)

Instrucciones o comandos para modificar (insertar, actualizar y eliminar) la información de una base de datos.

## DDL

Consulte [lenguaje de definición de bases de datos](#).

## conjunto profundo

Combinar varios modelos de aprendizaje profundo para la predicción. Puede utilizar conjuntos profundos para obtener una predicción más precisa o para estimar la incertidumbre de las predicciones.

## aprendizaje profundo

Un subcampo del ML que utiliza múltiples capas de redes neuronales artificiales para identificar el mapeo entre los datos de entrada y las variables objetivo de interés.

## defensa en profundidad

Un enfoque de seguridad de la información en el que se distribuyen cuidadosamente una serie de mecanismos y controles de seguridad en una red informática para proteger la confidencialidad, la integridad y la disponibilidad de la red y de los datos que contiene. Al adoptar esta estrategia AWS, se añaden varios controles en diferentes capas de la AWS Organizations estructura para ayudar a proteger los recursos. Por ejemplo, un enfoque de defensa en profundidad podría combinar la autenticación multifactor, la segmentación de la red y el cifrado.

## administrador delegado

En AWS Organizations, un servicio compatible puede registrar una cuenta de AWS miembro para administrar las cuentas de la organización y gestionar los permisos de ese servicio. Esta cuenta se denomina administrador delegado para ese servicio. Para obtener más información y una lista de servicios compatibles, consulte [Servicios que funcionan con AWS Organizations](#) en la documentación de AWS Organizations .

## Implementación

El proceso de hacer que una aplicación, características nuevas o correcciones de código se encuentren disponibles en el entorno de destino. La implementación abarca implementar cambios en una base de código y, a continuación, crear y ejecutar esa base en los entornos de la aplicación.

## entorno de desarrollo

Consulte [entorno](#).

## control de detección

Un control de seguridad que se ha diseñado para detectar, registrar y alertar después de que se produzca un evento. Estos controles son una segunda línea de defensa, ya que lo advierten sobre los eventos de seguridad que han eludido los controles preventivos establecidos. Para obtener más información, consulte [Controles de detección](#) en Implementación de controles de seguridad en AWS.

## asignación de flujos de valor para el desarrollo (DVSM)

Proceso que se utiliza para identificar y priorizar las restricciones que afectan negativamente a la velocidad y la calidad en el ciclo de vida del desarrollo de software. DVSM amplía el proceso de asignación del flujo de valor diseñado originalmente para las prácticas de fabricación ajustada. Se centra en los pasos y los equipos necesarios para crear y transferir valor a través del proceso de desarrollo de software.

## gemelo digital

Representación virtual de un sistema del mundo real, como un edificio, una fábrica, un equipo industrial o una línea de producción. Los gemelos digitales son compatibles con el mantenimiento predictivo, la supervisión remota y la optimización de la producción.

## tabla de dimensiones

En un [esquema en estrella](#), tabla más pequeña que contiene los atributos de datos sobre los datos cuantitativos en una tabla de hechos. Los atributos de la tabla de dimensiones suelen ser campos de texto o números discretos que se comportan como texto. Estos atributos se suelen utilizar para restringir consultas, filtrarlas y etiquetar los conjuntos de resultados.

## desastre

Un evento que impide que una carga de trabajo o un sistema cumplan sus objetivos empresariales en su ubicación principal de implementación. Estos eventos pueden ser desastres naturales, fallos técnicos o el resultado de acciones humanas, como una configuración incorrecta involuntaria o un ataque de malware.

## recuperación de desastres (DR)

Estrategia y proceso que utiliza para minimizar el tiempo de inactividad y la pérdida de datos a causa de un [desastre](#). Para obtener más información, consulte [Recuperación de cargas de trabajo ante desastres en AWS: Recuperación en la nube](#) en el AWS Well-Architected marco.

## DML

Consulte [lenguaje de manipulación de bases de datos](#).

## diseño basado en el dominio

Un enfoque para desarrollar un sistema de software complejo mediante la conexión de sus componentes a dominios en evolución, o a los objetivos empresariales principales, a los que sirve cada componente. Eric Evans introdujo este concepto en su libro *Domain-Driven Design: Tackling Complexity in the Heart of Software* (Boston: Addison-Wesley Professional, 2003). Para

obtener información sobre cómo utilizar el diseño basado en dominios con el patrón de higos estranguladores, consulte [Modernización gradual de los servicios web antiguos de ASP.NET Microsoft \(ASMX\) mediante contenedores y Amazon API Gateway](#).

## DR

Consulte [recuperación ante desastres](#).

### Detección de desviaciones

Seguimiento de las desviaciones con respecto a una configuración con línea de base. Por ejemplo, puedes usarlo AWS CloudFormation para [detectar desviaciones en los recursos del sistema](#) o puedes usarlo AWS Control Tower para [detectar cambios en tu landing zone](#) que puedan afectar al cumplimiento de los requisitos de gobierno.

## DVSM

Consulte [asignación de flujos de valor para el desarrollo](#).

## E

### EDA

Consulte [análisis de datos de tipo exploratorio](#).

### EDI

Consulte [intercambio electrónico de datos](#).

### computación en la periferia

La tecnología que aumenta la potencia de cálculo de los dispositivos inteligentes en la periferia de una red de IoT. En comparación con la [computación en la nube](#), la computación de periferia puede reducir la latencia de la comunicación y mejorar el tiempo de respuesta.

### intercambio electrónico de datos (EDI)

Intercambio automatizado de documentos comerciales entre organizaciones. Para más información, consulte [¿Qué es el intercambio electrónico de datos?](#)

### cifrado

Proceso de computación que transforma datos de texto plano, que son legibles por humanos, en texto cifrado.

## clave de cifrado

Cadena criptográfica de bits aleatorios que se genera mediante un algoritmo de cifrado. Las claves pueden variar en longitud y cada una se ha diseñado para ser impredecible y única.

## endianidad

El orden en el que se almacenan los bytes en la memoria del ordenador. Big-endian los sistemas almacenan primero el byte más significativo. Little-endian los sistemas almacenan primero el byte menos significativo.

## punto de conexión

Consulte [punto de conexión de servicio](#).

## servicio de punto de conexión

Servicio que puede alojar en una nube privada virtual (VPC) para compartir con otros usuarios. Puede crear un servicio de punto final con AWS PrivateLink entidades principales Cuentas de AWS o AWS Identity and Access Management (de IAM) y conceder permisos a ellas. Estas cuentas o entidades principales pueden conectarse a su servicio de punto de conexión de forma privada mediante la creación de puntos de conexión de VPC de interfaz. Para obtener más información, consulte [Creación de un servicio de punto de conexión](#) en la documentación de Amazon Virtual Private Cloud (Amazon VPC).

## planificación de recursos empresariales (ERP)

Sistema que automatiza y administra los procesos empresariales clave (como la contabilidad, [MES](#) y la administración de proyectos) de una empresa.

## cifrado de sobre

El proceso de cifrar una clave de cifrado con otra clave de cifrado. Para obtener más información, consulte el [cifrado de sobres](#) en la documentación de AWS Key Management Service (AWS KMS).

## entorno

Una instancia de una aplicación en ejecución. Los siguientes son los tipos de entornos más comunes en la computación en la nube:

- entorno de desarrollo: instancia de una aplicación en ejecución que solo se encuentra disponible para el equipo principal responsable del mantenimiento de la aplicación. Los

entornos de desarrollo se utilizan para probar los cambios antes de promocionarlos a los entornos superiores. Este tipo de entorno a veces se denomina entorno de prueba.

- entornos inferiores: todos los entornos de desarrollo de una aplicación, como los que se utilizan para las compilaciones y pruebas iniciales.
- entorno de producción: instancia de una aplicación en ejecución a la que pueden acceder los usuarios finales. En un CI/CD proceso, el entorno de producción es el último entorno de implementación.
- entornos superiores: todos los entornos a los que pueden acceder usuarios que no sean del equipo de desarrollo principal. Esto puede incluir un entorno de producción, entornos de preproducción y entornos para las pruebas de aceptación por parte de los usuarios.

## epopeya

En las metodologías ágiles, son categorías funcionales que ayudan a organizar y priorizar el trabajo. Las epopeyas brindan una descripción detallada de los requisitos y las tareas de implementación. Por ejemplo, las epopeyas AWS de seguridad de CAF incluyen la gestión de identidades y accesos, los controles de detección, la seguridad de la infraestructura, la protección de datos y la respuesta a incidentes. Para obtener más información sobre las epopeyas en la estrategia de migración de AWS , consulte la [Guía de implementación del programa](#).

## ERP

Consulte [planificación de recursos empresariales](#).

## análisis de datos de tipo exploratorio (EDA)

El proceso de analizar un conjunto de datos para comprender sus características principales. Se recopilan o agregan datos y, a continuación, se realizan las investigaciones iniciales para encontrar patrones, detectar anomalías y comprobar las suposiciones. El EDA se realiza mediante el cálculo de estadísticas resumidas y la creación de visualizaciones de datos.

## F

### tabla de hechos

Tabla central de un [esquema en estrella](#). Almacena datos cuantitativos sobre operaciones empresariales. Por lo general, una tabla de hechos contiene dos tipos de columnas: las que contienen medidas y las que contienen una clave externa para una tabla de dimensiones.

## Fail Fast

Filosofía que utiliza pruebas frecuentes e incrementales para reducir el ciclo de vida del desarrollo. Es una parte fundamental de los enfoques ágiles.

### límite de aislamiento de errores

En el Nube de AWS, un límite, como una zona de disponibilidad Región de AWS, un plano de control o un plano de datos, que limita el efecto de una falla y ayuda a mejorar la resiliencia de las cargas de trabajo. Para más información, consulte [AWS Fault Isolation Boundaries](#).

### rama de característica

Consulte [rama](#).

### características

Los datos de entrada que se utilizan para hacer una predicción. Por ejemplo, en un contexto de fabricación, las características pueden ser imágenes que se capturan periódicamente desde la línea de fabricación.

### importancia de las características

La importancia que tiene una característica para las predicciones de un modelo. Por lo general, esto se expresa como una puntuación numérica que se puede calcular mediante diversas técnicas, como las explicaciones aditivas de Shapley (SHAP) y los gradientes integrados. Para obtener más información, consulte [Interpretabilidad del modelo de aprendizaje automático](#) con AWS

### transformación de funciones

Optimizar los datos para el proceso de ML, lo que incluye enriquecer los datos con fuentes adicionales, escalar los valores o extraer varios conjuntos de información de un solo campo de datos. Esto permite que el modelo de ML se beneficie de los datos. Por ejemplo, si divide la fecha del “27 de mayo de 2021 00:15:37” en “jueves”, “mayo”, “2021” y “15”, puede ayudar al algoritmo de aprendizaje a aprender patrones matizados asociados a los diferentes componentes de los datos.

### peticiones con pocos pasos

Proporcionar a un [LLM](#) una pequeña cantidad de ejemplos que demuestren la tarea y el resultado deseado antes de pedirle que lleve a cabo una tarea similar. Esta técnica es una aplicación del aprendizaje contextual, en el que los modelos aprenden a partir de ejemplos (tomas) integrados en las instrucciones. Few-shot Las indicaciones pueden ser eficaces para tareas que requieren

un formato, un razonamiento o un conocimiento del dominio específicos. Consulte también [peticiones desde cero](#).

## FGAC

Consulte [control de acceso detallado](#).

### control de acceso preciso (FGAC)

El uso de varias condiciones que tienen por objetivo permitir o denegar una solicitud de acceso.

### migración relámpago

Método de migración de bases de datos que utiliza la replicación continua de datos mediante la [captura de datos de cambio](#) para migrar los datos en el menor tiempo posible, en lugar de utilizar un enfoque gradual. El objetivo es reducir al mínimo el tiempo de inactividad.

## FM

Consulte [modelo fundacional](#).

### Modelo fundacional (FM)

Gran red neuronal de aprendizaje profundo que se entrenó con conjuntos de datos masivos de datos generalizados y no etiquetados. Los FM pueden hacer una amplia variedad de tareas generales, como comprender el lenguaje, generar texto e imágenes y conversar en lenguaje natural. Para más información, consulte [¿Qué son los modelos fundacionales?](#)

### Puerta de enlace FM

Un intermediario centralizado que controla y normaliza el acceso a los modelos básicos. También se conoce como puerta de enlace LLM.

## G

### IA generativa

Subconjunto de modelos de [IA](#) que se entrenaron con grandes cantidades de datos y que pueden utilizar una simple petición de texto para crear contenido y artefactos nuevos, como imágenes, videos, texto y audio. Para más información, consulte [¿Qué es la IA generativa?](#)

### bloqueo geográfico

Consulte [restricciones geográficas](#).

## restricciones geográficas (bloqueo geográfico)

En Amazon CloudFront, una opción para impedir que los usuarios de países específicos accedan a las distribuciones de contenido. Puede utilizar una lista de permitidos o bloqueados para especificar los países aprobados y prohibidos. Para obtener más información, consulta [Restringir la distribución geográfica del contenido](#) en la CloudFront documentación.

## Flujo de trabajo de Gitflow

Un enfoque en el que los entornos inferiores y superiores utilizan diferentes ramas en un repositorio de código fuente. El flujo de trabajo de Gitflow se considera heredado, mientras que el [flujo de trabajo basado en enlaces troncales](#) es el enfoque moderno preferido.

## imagen dorada

Instantánea de un sistema o software que se usa como plantilla para implementar nuevas instancias de ese sistema o software. Por ejemplo, en la fabricación, una imagen dorada se puede utilizar para aprovisionar software en varios dispositivos y ayuda a mejorar la velocidad, la escalabilidad y la productividad de las operaciones de fabricación de dispositivos.

## estrategia de implementación desde cero

La ausencia de infraestructura existente en un entorno nuevo. Al adoptar una estrategia de implementación desde cero para una arquitectura de sistemas, puede seleccionar todas las tecnologías nuevas sin que estas deban ser compatibles con una infraestructura existente, lo que también se conoce como [implementación sobre infraestructura existente](#). Si está ampliando la infraestructura existente, puede combinar las estrategias de implementación sobre infraestructuras existentes y de implementación desde cero.

## barrera de protección

Una regla de alto nivel que ayuda a regular los recursos, las políticas y la conformidad en todas las unidades organizativas (OU). Las barreras de protección preventivas aplican políticas para garantizar la alineación con los estándares de conformidad. Se implementan mediante políticas de control de servicios y límites de permisos de IAM. Las barreras de protección de detección detectan las vulneraciones de las políticas y los problemas de conformidad, y generan alertas para su corrección. Se implementan mediante Amazon AWS Config AWS Security Hub CSPM GuardDuty AWS Trusted Advisor, Amazon Inspector y AWS Lambda cheques personalizados.

## barandas (AI)

Mecanismos de seguridad que filtran, validan y restringen las entradas y salidas de los [agentes](#) para ayudar a garantizar un comportamiento responsable y seguro de la IA.

# H

## HA

Consulte [alta disponibilidad](#).

### migración heterogénea de bases de datos

Migración de la base de datos de origen a una base de datos de destino que utilice un motor de base de datos diferente (por ejemplo, de Oracle a Amazon Aurora). La migración heterogénea suele ser parte de un esfuerzo de rediseño de la arquitectura y convertir el esquema puede ser una tarea compleja. [AWS ofrece AWS SCT](#), lo cual ayuda con las conversiones de esquemas.

### alta disponibilidad (HA)

La capacidad de una carga de trabajo para funcionar de forma continua, sin intervención, en caso de desafíos o desastres. Los sistemas de alta disponibilidad están diseñados para realizar una conmutación por error automática, ofrecer un rendimiento de alta calidad de forma constante y gestionar diferentes cargas y fallos con un impacto mínimo en el rendimiento.

### modernización histórica

Un enfoque utilizado para modernizar y actualizar los sistemas de tecnología operativa (TO) a fin de satisfacer mejor las necesidades de la industria manufacturera. Un histórico es un tipo de base de datos que se utiliza para recopilar y almacenar datos de diversas fuentes en una fábrica.

### datos de reserva

Parte de los datos históricos etiquetados que se ocultan de un conjunto de datos que se utiliza para entrenar un modelo de [machine learning](#). Puede utilizar los datos de reserva para evaluar el rendimiento del modelo mediante la comparación de las predicciones del modelo con los datos de reserva.

### human-in-the-loop (HiTL)

Un patrón de flujo de trabajo en el que la ejecución de los [agentes](#) se detiene para su revisión y aprobación por parte de una persona en los puntos de decisión críticos.

### migración homogénea de bases de datos

Migración de la base de datos de origen a una base de datos de destino que comparte el mismo motor de base de datos (por ejemplo, Microsoft SQL Server a Amazon RDS para SQL Server).

La migración homogénea suele formar parte de un esfuerzo para volver a alojar o redefinir la plataforma. Puede utilizar las utilidades de bases de datos nativas para migrar el esquema.

## datos recientes

Datos a los que se accede con frecuencia, como datos en tiempo real o datos traslacionales recientes. Por lo general, estos datos requieren un nivel o una clase de almacenamiento de alto rendimiento para proporcionar respuestas rápidas a las consultas.

## hotfix

Una solución urgente para un problema crítico en un entorno de producción. Debido a su urgencia, una revisión suele realizarse fuera del flujo de trabajo habitual de las DevOps versiones.

## periodo de hiperatención

Periodo, inmediatamente después de la transición, durante el cual un equipo de migración administra y monitorea las aplicaciones migradas en la nube para solucionar cualquier problema. Por lo general, este periodo dura de 1 a 4 días. Al final del periodo de hiperatención, el equipo de migración suele transferir la responsabilidad de las aplicaciones al equipo de operaciones en la nube.

## I

## laC

Consulte [infraestructura como código](#).

## políticas basadas en identidades

Política asociada a uno o más directores de IAM que define sus permisos en el entorno. Nube de AWS

## aplicación inactiva

Aplicación que utiliza un promedio de CPU y memoria de entre 5 y 20 por ciento durante un periodo de 90 días. En un proyecto de migración, es habitual retirar estas aplicaciones o mantenerlas en las instalaciones.

## IloT

Consulte [Internet de las cosas industrial](#).

## infraestructura inmutable

Modelo que implementa una nueva infraestructura para las cargas de trabajo de producción en lugar de actualizar o modificar la infraestructura existente o aplicarle revisiones. Las infraestructuras inmutables son de manera intrínseca más coherentes, fiables y predecibles que las [infraestructuras mutables](#). Para obtener más información, consulte las mejores prácticas del [Framework para implementar con una infraestructura inmutable](#). AWS Well-Architected

## VPC entrante (de entrada)

En una arquitectura de AWS cuentas múltiples, una VPC que acepta, inspecciona y enruta las conexiones de red desde fuera de una aplicación. La [Arquitectura de referencia de seguridad de AWS](#) recomienda configurar su cuenta de red con VPC entrantes, salientes y de inspección para proteger la interfaz bidireccional entre su aplicación e Internet en general.

## migración gradual

Estrategia de transición en la que se migra la aplicación en partes pequeñas en lugar de realizar una transición única y completa. Por ejemplo, puede trasladar inicialmente solo unos pocos microservicios o usuarios al nuevo sistema. Tras comprobar que todo funciona correctamente, puede trasladar microservicios o usuarios adicionales de forma gradual hasta que pueda retirar su sistema heredado. Esta estrategia reduce los riesgos asociados a las grandes migraciones.

## Industria 4.0

Un término que [Klaus Schwab](#) introdujo en 2016 para referirse a la modernización de los procesos de fabricación mediante avances en la conectividad, los datos en tiempo real, la automatización, el análisis y. AI/ML

## infraestructura

Todos los recursos y activos que se encuentran en el entorno de una aplicación.

## infraestructura como código (IaC)

Proceso de aprovisionamiento y administración de la infraestructura de una aplicación mediante un conjunto de archivos de configuración. La IaC se ha diseñado para ayudarlo a centralizar la administración de la infraestructura, estandarizar los recursos y escalar con rapidez a fin de que los entornos nuevos sean repetibles, fiables y consistentes.

## Internet de las cosas industrial (IIoT)

El uso de sensores y dispositivos conectados a Internet en los sectores industriales, como el productivo, el eléctrico, el automotriz, el sanitario, el de las ciencias de la vida y el de la

agricultura. Para obtener más información, consulte [Creación de una estrategia de transformación digital del Internet de las cosas industrial \(IIoT\)](#).

## VPC de inspección

En una arquitectura de AWS cuentas múltiples, una VPC centralizada que gestiona las inspecciones del tráfico de red entre las VPC (iguales o Regiones de AWS diferentes), Internet y las redes locales. La [Arquitectura de referencia de seguridad de AWS](#) recomienda configurar su cuenta de red con VPC entrantes, salientes y de inspección para proteger la interfaz bidireccional entre su aplicación e Internet en general.

## Internet de las cosas (IoT)

Red de objetos físicos conectados con sensores o procesadores integrados que se comunican con otros dispositivos y sistemas a través de Internet o de una red de comunicación local. Para obtener más información, consulte [¿Qué es IoT?](#).

## interpretabilidad

Característica de un modelo de machine learning que describe el grado en que un ser humano puede entender cómo las predicciones del modelo dependen de sus entradas. Para obtener más información, consulte Interpretabilidad del modelo [de aprendizaje automático](#) con AWS

## IoT

Consulte [Internet de las cosas](#).

## biblioteca de información de TI (ITIL)

Conjunto de prácticas recomendadas para ofrecer servicios de TI y alinearlos con los requisitos empresariales. La ITIL proporciona la base para la ITSM.

## administración de servicios de TI (ITSM)

Actividades asociadas con el diseño, la implementación, la administración y el soporte de los servicios de TI para una organización. Para obtener información sobre la integración de las operaciones en la nube con las herramientas de ITSM, consulte la [Guía de integración de operaciones](#).

## ITIL

Consulte [biblioteca de información de TI](#).

## ITSM

Consulte [administración de servicios de TI](#).

## L

### control de acceso basado en etiquetas (LBAC)

Una implementación del control de acceso obligatorio (MAC) en la que a los usuarios y a los propios datos se les asigna explícitamente un valor de etiqueta de seguridad. La intersección entre la etiqueta de seguridad del usuario y la etiqueta de seguridad de los datos determina qué filas y columnas puede ver el usuario.

### zona de aterrizaje

Una landing zone es un AWS entorno multicuenta bien diseñado, escalable y seguro. Este es un punto de partida desde el cual las empresas pueden lanzar e implementar rápidamente cargas de trabajo y aplicaciones con confianza en su entorno de seguridad e infraestructura. Para obtener más información sobre las zonas de aterrizaje, consulte [Configuración de un entorno de AWS seguro y escalable con varias cuentas](#).

### modelo de lenguaje de gran tamaño (LLM)

Modelo de [IA](#) de aprendizaje profundo que se entrenó previamente con una gran cantidad de datos. Un LLM puede llevar a cabo varias tareas, como responder preguntas, resumir documentos, traducir textos a otros idiomas y completar oraciones. Para más información, consulte [¿Qué es un LLM \(modelo de lenguaje de gran tamaño\)?](#)

### migración grande

Migración de 300 servidores o más.

### LBAC

Consulte [control de acceso basado en etiquetas](#).

### privilegio mínimo

La práctica recomendada de seguridad que consiste en conceder los permisos mínimos necesarios para realizar una tarea. Para obtener más información, consulte [Aplicar permisos de privilegio mínimo](#) en la documentación de IAM.

### migrar mediante lift-and-shift

Consulte [Las 7 R](#).

### sistema little-endian

Un sistema que almacena primero el byte menos significativo. Consulte también [endianidad](#).

## LLM

Consulte [modelo de lenguaje de gran tamaño](#).

entornos inferiores

Consulte [entorno](#).

## M

machine learning (ML)

Un tipo de inteligencia artificial que utiliza algoritmos y técnicas para el reconocimiento y el aprendizaje de patrones. El ML analiza y aprende de los datos registrados, como los datos del Internet de las cosas (IoT), para generar un modelo estadístico basado en patrones. Para más información, consulte [Machine learning](#).

rama principal

Consulte [rama](#).

malware

Software diseñado para comprometer la seguridad o la privacidad de la computadora. El malware podría interrumpir los sistemas informáticos, filtrar información confidencial u obtener acceso no autorizado. Algunos ejemplos de malware son los virus, los gusanos, el ransomware, los troyanos, el spyware y los registradores de pulsaciones de teclas.

Servicios administrados

Servicios de AWS en el que AWS opera la capa de infraestructura, el sistema operativo y las plataformas, y se accede a los puntos finales para almacenar y recuperar datos. Amazon Simple Storage Service (Amazon S3) y Amazon DynamoDB son ejemplos de servicios administrados. También se conocen como servicios abstractos.

sistema de ejecución de fabricación (MES)

Sistema de software para seguir, supervisar, documentar y controlar los procesos de producción que convierten las materias primas en productos acabados en la zona de producción.

MAP

Consulte [Programa de aceleración de la migración](#).

## MCP

Consulte [Model Context Protocol](#).

### Protocolo de contexto para modelos (MCP)

Un protocolo sin estado para la comunicación entre el [agente](#) y la [herramienta](#).

### Servidor MCP

Un servicio que expone una o más [herramientas](#) a través del protocolo [Model Context](#).

### mecanismo

Proceso completo mediante el que se crea una herramienta, se impulsa su adopción y, a continuación, se inspeccionan los resultados para hacer ajustes. Un mecanismo es un ciclo que se refuerza y mejora por sí mismo a medida que funciona. Para obtener más información, consulte [Creación de mecanismos](#) en el AWS Well-Architected marco.

### cuenta de miembro

Todas las Cuentas de AWS demás cuentas, excepto la de administración, que forman parte de una organización AWS Organizations. Una cuenta no puede pertenecer a más de una organización a la vez.

## MES

Consulte [sistema de ejecución de fabricación](#).

### Message Queuing Telemetry Transport (MQTT)

[Un protocolo de comunicación ligero de máquina a máquina \(M2M\), basado en el publish/subscribe patrón, para dispositivos de IoT con recursos limitados.](#)

### microservicio

Un servicio pequeño e independiente que se comunica a través de API bien definidas y que, por lo general, es propiedad de equipos pequeños e independientes. Por ejemplo, un sistema de seguros puede incluir microservicios que se adapten a las capacidades empresariales, como las de ventas o marketing, o a subdominios, como las de compras, reclamaciones o análisis. Los beneficios de los microservicios incluyen la agilidad, la escalabilidad flexible, la facilidad de implementación, el código reutilizable y la resiliencia. Para obtener más información, consulte [Integrar](#) microservicios mediante servicios sin servidor. AWS

## arquitectura de microservicios

Un enfoque para crear una aplicación con componentes independientes que ejecutan cada proceso de la aplicación como un microservicio. Estos microservicios se comunican a través de una interfaz bien definida mediante API ligeras. Cada microservicio de esta arquitectura se puede actualizar, implementar y escalar para satisfacer la demanda de funciones específicas de una aplicación. Para obtener más información, consulte [Implementación de microservicios](#) en AWS

## Programa de aceleración de la migración (MAP)

Un AWS programa que proporciona soporte de consultoría, formación y servicios para ayudar a las organizaciones a crear una base operativa sólida para migrar a la nube y para ayudar a compensar el costo inicial de las migraciones. El MAP incluye una metodología de migración para ejecutar las migraciones antiguas de forma metódica y un conjunto de herramientas para automatizar y acelerar los escenarios de migración más comunes.

## migración a escala

Proceso de transferencia de la mayoría de la cartera de aplicaciones a la nube en oleadas, con más aplicaciones desplazadas a un ritmo más rápido en cada oleada. En esta fase, se utilizan las prácticas recomendadas y las lecciones aprendidas en las fases anteriores para implementar una fábrica de migración de equipos, herramientas y procesos con el fin de agilizar la migración de las cargas de trabajo mediante la automatización y la entrega ágil. Esta es la tercera fase de la [estrategia de migración de AWS](#).

## fábrica de migración

Cross-functional equipos que agilizan la migración de las cargas de trabajo mediante enfoques ágiles y automatizados. Los equipos de las fábricas de migración suelen estar compuestos por analistas y propietarios de operaciones, ingenieros de migración, desarrolladores y DevOps profesionales que trabajan a pasos agigantados. Entre el 20 y el 50 por ciento de la cartera de aplicaciones empresariales se compone de patrones repetidos que pueden optimizarse mediante un enfoque de fábrica. Para obtener más información, consulte la [discusión sobre las fábricas de migración](#) y la [Guía de fábricas de migración a la nube](#) en este contenido.

## metadatos de migración

Información sobre la aplicación y el servidor que se necesita para completar la migración. Cada patrón de migración requiere un conjunto diferente de metadatos de migración. Algunos ejemplos de metadatos de migración son la subred de destino, el grupo de seguridad y AWS la cuenta.

## patrón de migración

Tarea de migración repetible que detalla la estrategia de migración, el destino de la migración y la aplicación o el servicio de migración utilizados. Ejemplo: rehospede la migración a Amazon EC2 AWS con Application Migration Service.

## Migration Portfolio Assessment (MPA)

Herramienta en línea que proporciona información a fin de validar los argumentos comerciales necesarios para migrar a la Nube de AWS. La MPA ofrece una evaluación detallada de la cartera (adecuación del tamaño de los servidores, precios, comparaciones del costo total de propiedad, análisis de los costos de migración), así como una planificación de la migración (análisis y recopilación de datos de aplicaciones, agrupación de aplicaciones, priorización de la migración y planificación de oleadas). La [herramienta MPA](#) (requiere iniciar sesión) está disponible de forma gratuita para todos los AWS consultores y consultores de los socios de APN.

## Evaluación de la preparación para la migración (MRA)

Proceso que consiste en obtener información sobre el estado de preparación de una organización para la nube, identificar sus puntos fuertes y débiles y elaborar un plan de acción para cerrar las brechas identificadas mediante el AWS CAF. Para obtener más información, consulte la [Guía de preparación para la migración](#). La MRA es la primera fase de la [estrategia de migración de AWS](#).

## estrategia de migración

Enfoque utilizado para migrar una carga de trabajo a la Nube de AWS. Para más información, consulte la entrada [Las 7 R](#) de este glosario y también [Mobilize your organization to accelerate large-scale migrations](#).

## ML

Consulte [machine learning](#).

## modernización

Transformar una aplicación obsoleta (antigua o monolítica) y su infraestructura en un sistema ágil, elástico y de alta disponibilidad en la nube para reducir los gastos, aumentar la eficiencia y aprovechar las innovaciones. Para más información, consulte [Strategy for modernizing applications in the Nube de AWS](#).

## evaluación de la preparación para la modernización

Evaluación que ayuda a determinar la preparación para la modernización de las aplicaciones de una organización; identifica los beneficios, los riesgos y las dependencias; y determina qué

tan bien la organización puede soportar el estado futuro de esas aplicaciones. El resultado de la evaluación es un esquema de la arquitectura objetivo, una hoja de ruta que detalla las fases de desarrollo y los hitos del proceso de modernización y un plan de acción para abordar las brechas identificadas. Para más información, consulte [Evaluating modernization readiness for applications in the Nube de AWS](#).

#### aplicaciones monolíticas (monolitos)

Aplicaciones que se ejecutan como un único servicio con procesos estrechamente acoplados. Las aplicaciones monolíticas presentan varios inconvenientes. Si una característica de la aplicación experimenta un aumento en la demanda, se debe escalar toda la arquitectura. Agregar o mejorar las características de una aplicación monolítica también se vuelve más complejo a medida que crece la base de código. Para solucionar problemas con la aplicación, puede utilizar una arquitectura de microservicios. Para obtener más información, consulte [Descomposición de monolitos en microservicios](#).

#### MPA

Consulte [Migration Portfolio Assessment](#).

#### MQTT

Consulte [Message Queuing Telemetry Transport](#).

#### clasificación multiclase

Un proceso que ayuda a generar predicciones para varias clases (predice uno de más de dos resultados). Por ejemplo, un modelo de ML podría preguntar “¿Este producto es un libro, un automóvil o un teléfono?” o “¿Qué categoría de productos es más interesante para este cliente?”.

#### infraestructura mutable

Modelo que actualiza y modifica la infraestructura actual para las cargas de trabajo de producción. Para mejorar la coherencia, la confiabilidad y la previsibilidad, el AWS Well-Architected Marco recomienda el uso de una [infraestructura inmutable](#) como práctica recomendada.

## O

#### OAC

Consulte [control de acceso de origen](#).

## OAI

Consulte [identidad de acceso de origen](#).

## OCM

Consulte [administración del cambio organizacional](#).

### migración fuera de línea

Método de migración en el que la carga de trabajo de origen se elimina durante el proceso de migración. Este método implica un tiempo de inactividad prolongado y, por lo general, se utiliza para cargas de trabajo pequeñas y no críticas.

## OI

Consulte [integración de operaciones](#).

## OLA

Consulte [acuerdo de nivel operativo](#).

### migración en línea

Método de migración en el que la carga de trabajo de origen se copia al sistema de destino sin que se desconecte. Las aplicaciones que están conectadas a la carga de trabajo pueden seguir funcionando durante la migración. Este método implica un tiempo de inactividad nulo o mínimo y, por lo general, se utiliza para cargas de trabajo de producción críticas.

## OPC-UA

Consulte [Open Process Communications: arquitectura unificada](#).

### Comunicaciones de proceso abierto: arquitectura unificada () OPC-UA

Un protocolo de comunicación de máquina a máquina (M2M) para la automatización industrial. OPC-UA proporciona un estándar de interoperabilidad con esquemas de cifrado, autenticación y autorización de datos.

### acuerdo de nivel operativo (OLA)

Acuerdo que aclara lo que los grupos de TI operativos se comprometen a ofrecerse entre sí, para respaldar un acuerdo de nivel de servicio (SLA).

### revisión de la preparación operativa (ORR)

Lista de comprobación de preguntas y prácticas recomendadas asociadas que son útiles para comprender, evaluar, prevenir o reducir el alcance de los incidentes y posibles errores. Para

obtener más información, consulte [las revisiones de preparación operativa \(ORR\)](#) en el AWS Well-Architected marco.

### tecnología operativa (TO)

Sistemas de hardware y software que funcionan con el entorno físico para controlar las operaciones, los equipos y la infraestructura industriales. En el sector de la fabricación, la integración de los sistemas de TO y tecnología de la información (TI) es un enfoque clave para las transformaciones de la [industria 4.0](#).

### integración de operaciones (OI)

Proceso de modernización de las operaciones en la nube, que implica la planificación de la preparación, la automatización y la integración. Para obtener más información, consulte la [Guía de integración de las operaciones](#).

### registro de seguimiento organizativo

Un registro creado por y AWS CloudTrail que registra todos los eventos Cuentas de AWS de una organización AWS Organizations. Este registro de seguimiento se crea en cada Cuenta de AWS que forma parte de la organización y realiza un seguimiento de la actividad en cada cuenta. Para obtener más información, consulte [Crear un registro para una organización](#) en la CloudTrail documentación.

### administración del cambio organizacional (OCM)

Marco para administrar las transformaciones empresariales importantes y disruptivas desde la perspectiva de las personas, la cultura y el liderazgo. La OCM ayuda a las empresas a prepararse para nuevos sistemas y estrategias y a realizar la transición a ellos, al acelerar la adopción de cambios, abordar los problemas de transición e impulsar cambios culturales y organizacionales. En la estrategia de AWS migración, este marco se denomina aceleración de personal, debido a la velocidad de cambio que requieren los proyectos de adopción de la nube. Para obtener más información, consulte la [Guía de OCM](#).

### control de acceso de origen (OAC)

En CloudFront, una opción mejorada para restringir el acceso y proteger el contenido del Amazon Simple Storage Service (Amazon S3). El OAC admite todos los buckets de S3 Regiones de AWS, el cifrado del lado del servidor con AWS KMS (SSE-KMS) y DELETE las solicitudes PUT y dinámicas al bucket de S3.

## identidad de acceso de origen (OAI)

En CloudFront, una opción para restringir el acceso y proteger el contenido de Amazon S3. Cuando utiliza OAI, CloudFront crea un principal con el que Amazon S3 puede autenticarse. Los directores autenticados solo pueden acceder al contenido de un bucket de S3 a través de una distribución específica. CloudFront Consulte también el [OAC](#), que proporciona un control de acceso más detallado y mejorado.

## ORR

Consulte [revisión de la preparación operativa](#).

## OT

Consulte [tecnología operativa](#).

## VPC saliente (de salida)

En una arquitectura de AWS cuentas múltiples, una VPC que gestiona las conexiones de red que se inician desde una aplicación. La [Arquitectura de referencia de seguridad de AWS](#) recomienda configurar su cuenta de red con VPC entrantes, salientes y de inspección para proteger la interfaz bidireccional entre su aplicación e Internet en general.

## P

### límite de permisos

Una política de administración de IAM que se adjunta a las entidades principales de IAM para establecer los permisos máximos que puede tener el usuario o el rol. Para obtener más información, consulte [Límites de permisos](#) en la documentación de IAM.

### información de identificación personal (PII)

Información que, vista directamente o combinada con otros datos relacionados, puede utilizarse para deducir de manera razonable la identidad de una persona. Algunos ejemplos de información de identificación personal son los nombres, las direcciones y la información de contacto.

## PII

Consulte [información de identificación personal](#).

## manual de estrategias

Conjunto de pasos predefinidos que capturan el trabajo asociado a las migraciones, como la entrega de las funciones de operaciones principales en la nube. Un manual puede adoptar la forma de scripts, manuales de procedimientos automatizados o resúmenes de los procesos o pasos necesarios para operar un entorno modernizado.

## PLC

Consulte [controlador lógico programable](#).

## PLM

Consulte [administración del ciclo de vida del producto](#).

## policy

Objeto que puede definir permisos (consulte [política basada en identidad](#)), especificar las condiciones de acceso (consulte [política basada en recursos](#)) o definir los permisos máximos para todas las cuentas de una organización de AWS Organizations (consulte [política de control de servicio](#)).

## persistencia políglota

Elegir de forma independiente la tecnología de almacenamiento de datos de un microservicio en función de los patrones de acceso a los datos y otros requisitos. Si sus microservicios tienen la misma tecnología de almacenamiento de datos, pueden enfrentarse a desafíos de implementación o experimentar un rendimiento deficiente. Los microservicios se implementan más fácilmente y logran un mejor rendimiento y escalabilidad si utilizan el almacén de datos que mejor se adapte a sus necesidades.

## evaluación de cartera

Proceso de detección, análisis y priorización de la cartera de aplicaciones para planificar la migración. Para obtener más información, consulte la [Evaluación de la preparación para la migración](#).

## predicate

Condición de consulta que devuelve true o false. En general, se encuentra en una cláusula WHERE.

## inserción de predicados

Técnica de optimización de consultas en bases de datos que filtra los datos de la consulta antes de transferirlos. Esta técnica reduce la cantidad de datos de la base de datos relacional que se tienen que recuperar y procesar. Además, mejora el rendimiento de las consultas.

## control preventivo

Un control de seguridad diseñado para evitar que ocurra un evento. Estos controles son la primera línea de defensa para evitar el acceso no autorizado o los cambios no deseados en la red. Para obtener más información, consulte [Controles preventivos](#) en Implementación de controles de seguridad en AWS.

## entidad principal

Una entidad AWS que puede realizar acciones y acceder a los recursos. Esta entidad suele ser un usuario raíz para un Cuenta de AWS rol de IAM o un usuario. Para obtener más información, consulte Entidad principal en [Términos y conceptos de roles](#) en la documentación de IAM.

## Privacidad desde el diseño

Enfoque de ingeniería de sistemas que tiene en cuenta la privacidad durante todo el proceso de desarrollo.

## zonas alojadas privadas

Contenedor que aloja información acerca de cómo desea que responda Amazon Route 53 a las consultas de DNS de un dominio y sus subdominios en una o varias VPC. Para obtener más información, consulte [Uso de zonas alojadas privadas](#) en la documentación de Route 53.

## control proactivo

[Control de seguridad](#) que se diseñó para evitar la implementación de recursos que no cumplan con la normativa. Estos controles analizan los recursos antes de aprovisionarlos. Si el recurso no cumple con los requisitos del control, no se aprovisiona. Para obtener más información, consulte la [guía de referencia de controles](#) en la AWS Control Tower documentación y consulte [Controles proactivos](#) en Implementación de controles de seguridad en AWS.

## administración del ciclo de vida del producto (PLM)

Administración de los datos y los procesos de un producto a lo largo de todo su ciclo de vida, desde el diseño, el desarrollo y el lanzamiento, pasando por el crecimiento y la madurez, hasta la reducción de su uso y su retirada.

## entorno de producción

Consulte [entorno](#).

## controlador lógico programable (PLC)

En el sector de la fabricación, computadora adaptable y altamente fiable que supervisa las máquinas y automatiza los procesos de fabricación.

## encadenamiento de peticiones

Uso de la salida de una petición de [LLM](#) como entrada para la siguiente petición a fin de generar mejores respuestas. Esta técnica se utiliza para dividir una tarea compleja en tareas secundarias o para refinar o ampliar de forma iterativa una respuesta preliminar. Ayuda a mejorar la precisión y la relevancia de las respuestas de un modelo y permite obtener resultados más detallados y personalizados.

## seudonimización

El proceso de reemplazar los identificadores personales de un conjunto de datos por valores de marcadores de posición. La seudonimización puede ayudar a proteger la privacidad personal. Los datos seudonimizados siguen considerándose datos personales.

## publish/subscribe (pub/sub)

Patrón que permite establecer comunicaciones asíncronas entre microservicios para mejorar la escalabilidad y la capacidad de respuesta. Por ejemplo, en un [MES](#) basado en microservicios, un microservicio puede publicar mensajes de eventos en un canal al que se pueden suscribir otros microservicios. El sistema puede agregar nuevos microservicios sin cambiar el servicio de publicación.

## Q

### plan de consulta

Serie de pasos, como instrucciones, que se utilizan para acceder a los datos de un sistema de base de datos relacional SQL.

### regresión del plan de consulta

El optimizador de servicios de la base de datos elige un plan menos óptimo que antes de un cambio determinado en el entorno de la base de datos. Los cambios en estadísticas,

restricciones, configuración del entorno, enlaces de parámetros de consultas y actualizaciones del motor de base de datos PostgreSQL pueden provocar una regresión del plan.

## R

### Matriz RACI

Consulte [responsable, fiable, consultada e informada \(RACI\)](#).

### RAG

Consulte [generación aumentada por recuperación](#).

### ransomware

Software malicioso que se ha diseñado para bloquear el acceso a un sistema informático o a los datos hasta que se efectúe un pago.

### Matriz RASCI

Consulte [responsable, fiable, consultada e informada \(RACI\)](#).

### RCAC

Consulte [control de acceso por filas y columnas](#).

### réplica de lectura

Una copia de una base de datos que se utiliza con fines de solo lectura. Puede enrutar las consultas a la réplica de lectura para reducir la carga en la base de datos principal.

### rediseñar

Consulte [Las 7 R](#).

### objetivo de punto de recuperación (RPO)

La cantidad de tiempo máximo aceptable desde el último punto de recuperación de datos. Esto determina qué se considera una pérdida de datos aceptable entre el último punto de recuperación y la interrupción del servicio.

### objetivo de tiempo de recuperación (RTO)

La demora máxima aceptable entre la interrupción del servicio y el restablecimiento del servicio.

## refactorizar

Consulte [Las 7 R.](#)

## Region

Conjunto de AWS recursos en un área geográfica. Cada uno Región de AWS está aislado e independiente de los demás para proporcionar tolerancia a las fallas, estabilidad y resiliencia. Para más información, consulte [Specify which Regions de AWS your account can use.](#)

## regresión

Una técnica de ML que predice un valor numérico. Por ejemplo, para resolver el problema de “¿A qué precio se venderá esta casa?”, un modelo de ML podría utilizar un modelo de regresión lineal para predecir el precio de venta de una vivienda en función de datos conocidos sobre ella (por ejemplo, los metros cuadrados).

## volver a alojar

Consulte [Las 7 R.](#)

## versión

En un proceso de implementación, el acto de promover cambios en un entorno de producción.

## reubicar

Consulte [Las 7 R.](#)

## redefinir la plataforma

Consulte [Las 7 R.](#)

## recomprar

Consulte [Las 7 R.](#)

## resiliencia

Capacidad de una aplicación para resistir interrupciones o recuperarse de ellas. Al planificar la resiliencia en la Nube de AWS, la [alta disponibilidad](#) y la [recuperación ante desastres](#) son consideraciones comunes. Para más información, consulte [Resiliencia en la Nube de AWS.](#)

## política basada en recursos

Una política asociada a un recurso, como un bucket de Amazon S3, un punto de conexión o una clave de cifrado. Este tipo de política especifica a qué entidades principales se les permite el acceso, las acciones compatibles y cualquier otra condición que deba cumplirse.

## matriz responsable, confiable, consultada e informada (RACI)

Una matriz que define las funciones y responsabilidades de todas las partes involucradas en las actividades de migración y las operaciones de la nube. El nombre de la matriz se deriva de los tipos de responsabilidad definidos en la matriz: responsable (R), contable (A), consultado (C) e informado (I). El tipo de soporte (S) es opcional. Si incluye el soporte, la matriz se denomina matriz RASCI y, si la excluye, se denomina matriz RACI.

## control receptivo

Un control de seguridad que se ha diseñado para corregir los eventos adversos o las desviaciones con respecto a su base de seguridad. Para obtener más información, consulte [Controles receptivos](#) en Implementación de controles de seguridad en AWS.

## retain

Consulte [Las 7 R](#).

## retirar

Consulte [Las 7 R](#).

## Generación aumentada de recuperación (RAG)

Tecnología de [IA generativa](#) mediante la que un [LLM](#) hace referencia a un origen de datos autorizado que se encuentra fuera de sus orígenes de datos de entrenamiento antes de generar una respuesta. Por ejemplo, un modelo de RAG podría hacer una búsqueda semántica en la base de conocimientos o en los datos personalizados de una organización. Para más información, consulte [¿Qué es RAG \(generación aumentada por recuperación\)?](#)

## rotación

Proceso mediante el que periódicamente se actualiza un [secreto](#) para que resulte más difícil que un atacante pueda acceder a las credenciales.

## control de acceso por filas y columnas (RCAC)

El uso de expresiones SQL básicas y flexibles que tienen reglas de acceso definidas. El RCAC consta de permisos de fila y máscaras de columnas.

## RPO

Consulte [objetivo de punto de recuperación](#).

## RTO

Consulte [objetivo de tiempo de recuperación](#).

## manual de procedimientos

Conjunto de procedimientos manuales o automatizados necesarios para realizar una tarea específica. Por lo general, se diseñan para agilizar las operaciones o los procedimientos repetitivos con altas tasas de error.

## S

### SAML 2.0

Un estándar abierto que utilizan muchos proveedores de identidad (IdPs). Esta función permite el inicio de sesión único (SSO) federado, de modo que los usuarios pueden iniciar sesión Consola de administración de AWS o llamar a las operaciones de la AWS API sin tener que crear un usuario en IAM para todos los miembros de la organización. Para obtener más información sobre la federación basada en SAML 2.0, consulte [Acerca de la federación basada en SAML 2.0](#) en la documentación de IAM.

### SCADA

Consulte [control de supervisión y adquisición de datos](#).

### SCP

Consulte [política de control de servicio](#).

### secreta

En AWS Secrets Manager, información confidencial o restringida, como una contraseña o credenciales de usuario, que se almacena de forma cifrada. Se compone del valor del secreto y de sus metadatos. El valor del secreto puede ser binario, una sola cadena o varias cadenas. Para más información, consulte [What's in a Secrets Manager secret?](#) en la documentación de Secrets Manager.

### seguridad desde el diseño

Enfoque de ingeniería de sistemas que tiene en cuenta la seguridad durante todo el proceso de desarrollo.

### control de seguridad

Barrera de protección técnica o administrativa que impide, detecta o reduce la capacidad de un agente de amenazas para aprovechar una vulnerabilidad de seguridad. Existen cuatro tipos de controles de seguridad principales: [preventivos](#), [de detección](#), [de respuesta](#) y [proactivos](#).

## refuerzo de la seguridad

Proceso de reducir la superficie expuesta a ataques para hacerla más resistente a los ataques. Esto puede incluir acciones, como la eliminación de los recursos que ya no se necesitan, la implementación de prácticas recomendadas de seguridad consistente en conceder privilegios mínimos o la desactivación de características innecesarias en los archivos de configuración.

## sistema de información sobre seguridad y administración de eventos (SIEM)

Herramientas y servicios que combinan sistemas de administración de información sobre seguridad (SIM) y de administración de eventos de seguridad (SEM). Un sistema de SIEM recopila, monitorea y analiza los datos de servidores, redes, dispositivos y otras fuentes para detectar amenazas y brechas de seguridad y generar alertas.

## automatización de la respuesta de seguridad

Acción predefinida y programada que está diseñada para responder automáticamente a un evento de seguridad o corregirlo. Estas automatizaciones sirven como controles de seguridad [preventivos o adaptables](#) que le ayudan a implementar las mejores prácticas AWS de seguridad. La modificación de un grupo de seguridad de VPC, la aplicación de revisiones a una instancia de Amazon EC2 o la rotación de credenciales son algunos ejemplos de acciones de respuesta automatizadas.

## cifrado del servidor

Cifrado de los datos en su destino, por parte de Servicio de AWS quien los recibe.

## política de control de servicio (SCP)

Una política que proporciona un control centralizado de los permisos de todas las cuentas de una organización en AWS Organizations. Las SCP definen barreras de protección o establecen límites a las acciones que un administrador puede delegar en los usuarios o roles. Puede utilizar las SCP como listas de permitidos o rechazados, para especificar qué servicios o acciones se encuentra permitidos o prohibidos. Para obtener más información, consulte [las políticas de control del servicio](#) en la AWS Organizations documentación.

## punto de enlace de servicio

La URL del punto de entrada de un Servicio de AWS. Para conectarse mediante programación a un servicio de destino, puede utilizar un punto de conexión. Para obtener más información, consulte [Puntos de conexión de Servicio de AWS](#) en Referencia general de AWS.

## acuerdo de nivel de servicio (SLA)

Acuerdo que aclara lo que un equipo de TI se compromete a ofrecer a los clientes, como el tiempo de actividad y el rendimiento del servicio.

## indicador de nivel de servicio (SLI)

Medición de un aspecto del rendimiento de un servicio, como la tasa de errores, la disponibilidad o el rendimiento.

## objetivo de nivel de servicio (SLO)

Métrica objetivo que representa el estado de un servicio medido mediante un [indicador de nivel de servicio](#).

## modelo de responsabilidad compartida

Un modelo que describe la responsabilidad con AWS la que compartes la seguridad y el cumplimiento de la nube. AWS es responsable de la seguridad de la nube, mientras que usted es responsable de la seguridad en la nube. Para obtener más información, consulte el [Modelo de responsabilidad compartida](#).

## Shadow AI

Aplicaciones de [IA](#) no autorizadas creadas o utilizadas fuera de los canales regulados dentro de una organización.

## SIEM

Consulte [sistema de administración de eventos e información de seguridad](#).

## único punto de error (SPOF)

Error en un único componente crítico de una aplicación que puede interrumpir el sistema.

## SLA

Consulte [acuerdo de nivel de servicio](#).

## SLI

Consulte [indicador de nivel de servicio](#).

## SLO

Consulte [objetivo de nivel de servicio](#).

## modelo de dividir y sembrar

Un patrón para escalar y acelerar los proyectos de modernización. A medida que se definen las nuevas funciones y los lanzamientos de los productos, el equipo principal se divide para crear nuevos equipos de productos. Esto ayuda a ampliar las capacidades y los servicios de su organización, mejora la productividad de los desarrolladores y apoya la innovación rápida. Para más información, consulte [Phased approach to modernizing applications in the Nube de AWS](#).

## SPOF

Consulte [único punto de error](#).

## esquema en estrella

Estructura organizativa de una base de datos que utiliza una tabla de hechos de gran tamaño para almacenar datos transaccionales o medidos y una o varias tablas dimensionales más pequeñas para almacenar los atributos de los datos. Esta estructura está diseñada para utilizarse en un [almacén de datos](#) o con fines de inteligencia empresarial.

## patrón de higo estrangulador

Un enfoque para modernizar los sistemas monolíticos mediante la reescritura y el reemplazo gradual de las funciones del sistema hasta que se pueda dismantelar el sistema heredado. Este patrón utiliza la analogía de una higuera que crece hasta convertirse en un árbol estable y, finalmente, se apodera y reemplaza a su host. El patrón fue [presentado por Martin Fowler](#) como una forma de gestionar el riesgo al reescribir sistemas monolíticos. Para ver un ejemplo de cómo aplicar este patrón, consulte [Modernización gradual de los servicios web antiguos de Microsoft ASP.NET \(ASMX\) mediante contenedores y Amazon API Gateway](#).

## subred

Un intervalo de direcciones IP en la VPC. Una subred debe residir en una sola zona de disponibilidad.

## control de supervisión y adquisición de datos (SCADA)

En el sector de la fabricación, sistema que utiliza hardware y software para supervisar los activos físicos y las operaciones de producción.

## cifrado simétrico

Un algoritmo de cifrado que utiliza la misma clave para cifrar y descifrar los datos.

## pruebas sintéticas

Prueba de un sistema de manera que simule las interacciones de los usuarios para detectar posibles problemas o supervisar el rendimiento. Puede usar [Amazon CloudWatch Synthetics](#) para crear estas pruebas.

## petición del sistema

Técnica para proporcionar contexto, instrucciones o pautas a un [LLM](#) para dirigir su comportamiento. Las peticiones del sistema ayudan a establecer el contexto y las reglas para las interacciones con los usuarios.

## T

### etiquetas

Key-value pares que actúan como metadatos para organizar sus AWS recursos. Las etiquetas pueden ayudar a administrar, identificar, organizar, buscar y filtrar recursos de . Para obtener más información, consulte [Etiquetado de los recursos de AWS](#).

### variable de destino

El valor que intenta predecir en el ML supervisado. Esto también se conoce como variable de resultado. Por ejemplo, en un entorno de fabricación, la variable objetivo podría ser un defecto del producto.

### lista de tareas

Herramienta que se utiliza para hacer un seguimiento del progreso mediante un manual de procedimientos. La lista de tareas contiene una descripción general del manual de procedimientos y una lista de las tareas generales que deben completarse. Para cada tarea general, se incluye la cantidad estimada de tiempo necesario, el propietario y el progreso.

### entorno de prueba

Consulte [entorno](#).

### entrenamiento

Proporcionar datos de los que pueda aprender su modelo de ML. Los datos de entrenamiento deben contener la respuesta correcta. El algoritmo de aprendizaje encuentra patrones en los

datos de entrenamiento que asignan los atributos de los datos de entrada al destino (la respuesta que desea predecir). Genera un modelo de ML que captura estos patrones. Luego, el modelo de ML se puede utilizar para obtener predicciones sobre datos nuevos para los que no se conoce el destino.

## herramienta

Una función o API que un [agente](#) puede invocar para realizar operaciones en sistemas externos.

## puerta de enlace de tránsito

Centro de tránsito de red que puede utilizar para interconectar las VPC y las redes en las instalaciones. Para obtener más información, consulte [Qué es una pasarela de tránsito](#) en la AWS Transit Gateway documentación.

## flujo de trabajo basado en enlaces troncales

Un enfoque en el que los desarrolladores crean y prueban características de forma local en una rama de característica y, a continuación, combinan esos cambios en la rama principal. Luego, la rama principal se adapta a los entornos de desarrollo, preproducción y producción, de forma secuencial.

## acceso de confianza

Otorgar permisos a un servicio que especifique para realizar tareas en su organización AWS Organizations y en sus cuentas en su nombre. El servicio de confianza crea un rol vinculado al servicio en cada cuenta, cuando ese rol es necesario, para realizar las tareas de administración por usted. Para obtener más información, consulte [AWS Organizations Utilización con otros AWS servicios](#) en la AWS Organizations documentación.

## ajuste

Cambiar aspectos de su proceso de formación a fin de mejorar la precisión del modelo de ML. Por ejemplo, puede entrenar el modelo de ML al generar un conjunto de etiquetas, incorporar etiquetas y, luego, repetir estos pasos varias veces con diferentes ajustes para optimizar el modelo.

## equipo de dos pizzas

Un DevOps equipo pequeño al que puedes alimentar con dos pizzas. Un equipo formado por dos integrantes garantiza la mejor oportunidad posible de colaboración en el desarrollo de software.

## U

### incertidumbre

Un concepto que hace referencia a información imprecisa, incompleta o desconocida que puede socavar la fiabilidad de los modelos predictivos de ML. Hay dos tipos de incertidumbre: la incertidumbre epistémica se debe a datos limitados e incompletos, mientras que la incertidumbre aleatoria se debe al ruido y la aleatoriedad inherentes a los datos.

### tareas indiferenciadas

También conocido como tareas arduas, es el trabajo que es necesario para crear y operar una aplicación, pero que no proporciona un valor directo al usuario final ni proporciona una ventaja competitiva. Algunos ejemplos de tareas indiferenciadas son la adquisición, el mantenimiento y la planificación de la capacidad.

### entornos superiores

Consulte [entorno](#).

## V

### succión

Una operación de mantenimiento de bases de datos que implica limpiar después de las actualizaciones incrementales para recuperar espacio de almacenamiento y mejorar el rendimiento.

### control de versión

Procesos y herramientas que realizan un seguimiento de los cambios, como los cambios en el código fuente de un repositorio.

### Emparejamiento de VPC

Conexión entre dos VPC que permite enrutar el tráfico mediante direcciones IP privadas. Para obtener más información, consulte [¿Qué es una interconexión de VPC?](#) en la documentación de Amazon VPC.

### vulnerabilidad

Defecto de software o hardware que pone en peligro la seguridad del sistema.

## W

### caché caliente

Un búfer caché que contiene datos actuales y relevantes a los que se accede con frecuencia. La instancia de base de datos puede leer desde la caché del búfer, lo que es más rápido que leer desde la memoria principal o el disco.

### datos templados

Datos a los que el acceso es infrecuente. Al consultar este tipo de datos, normalmente se aceptan consultas moderadamente lentas.

### función de ventana

Función SQL que hace un cálculo en un grupo de filas que se relacionan de alguna manera con el registro actual. Las funciones de ventana son útiles para las tareas de procesamiento, como calcular una media móvil o acceder al valor de las filas en función de la posición relativa de la fila actual.

### carga de trabajo

Conjunto de recursos y código que ofrece valor comercial, como una aplicación orientada al cliente o un proceso de backend.

### flujo de trabajo

Grupos funcionales de un proyecto de migración que son responsables de un conjunto específico de tareas. Cada flujo de trabajo es independiente, pero respalda a los demás flujos de trabajo del proyecto. Por ejemplo, el flujo de trabajo de la cartera es responsable de priorizar las aplicaciones, planificar las oleadas y recopilar los metadatos de migración. El flujo de trabajo de la cartera entrega estos recursos al flujo de trabajo de migración, que luego migra los servidores y las aplicaciones.

### WORM

Consulte [escritura única y lectura múltiple](#).

### WQF

Consulte [AWS Workload Qualification Framework](#).

## escritura única y lectura múltiple (WORM)

Modelo de almacenamiento que escribe los datos una sola vez y evita que se eliminen o modifiquen. Los usuarios autorizados pueden leer los datos tantas veces como sea necesario, pero no los pueden cambiar. Esta infraestructura de almacenamiento de datos se considera [inmutable](#).

## Z

### ataque de día cero

Ataque, normalmente de malware, que se aprovecha de una [vulnerabilidad de día cero](#).

### vulnerabilidad de día cero

Un defecto o una vulnerabilidad sin mitigación en un sistema de producción. Los agentes de amenazas pueden usar este tipo de vulnerabilidad para atacar el sistema. Los desarrolladores suelen darse cuenta de la vulnerabilidad a raíz del ataque.

### peticiones desde cero

Proporcionar a un [LLM](#) instrucciones para llevar a cabo una tarea, pero sin ejemplos (pasos) que puedan ayudar a guiarlo. El LLM debe usar los conocimientos del entrenamiento previo para llevar a cabo la tarea. La eficacia de la petición desde cero depende de la complejidad de la tarea y de la calidad de la petición. Consulte también [peticiones con pocos pasos](#).

### aplicación zombi

Aplicación que utiliza un promedio de CPU y memoria menor al 5 por ciento. En un proyecto de migración, es habitual retirar estas aplicaciones.

Las traducciones son generadas a través de traducción automática. En caso de conflicto entre la traducción y la versión original de inglés, prevalecerá la versión en inglés.