



Integrazione di microservizi utilizzando servizi serverless AWS

# AWS Linee guida prescrittive



# AWS Linee guida prescrittive: Integrazione di microservizi utilizzando servizi serverless AWS

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

I marchi e l'immagine commerciale di Amazon non possono essere utilizzati in relazione a prodotti o servizi che non siano di Amazon, in una qualsiasi modalità che possa causare confusione tra i clienti o in una qualsiasi modalità che denigri o discrediti Amazon. Tutti gli altri marchi non di proprietà di Amazon sono di proprietà dei rispettivi proprietari, che possono o meno essere affiliati, collegati o sponsorizzati da Amazon.

---

# Table of Contents

Introduzione .....	1
Destinatari principali .....	1
Obiettivi .....	2
Sicurezza .....	2
Modelli di comunicazione .....	3
Comunicazione sincrona .....	3
Comunicazione asincrona .....	5
Spara e dimentica .....	5
Verifica dei reclami .....	6
Richiamata .....	7
Comunicazione bidirezionale .....	8
Opzioni di coordinamento .....	11
Orchestrazione .....	11
Esempio: Step Functions .....	11
Esempio: Amazon MWAA .....	14
Principali differenze tra Step Functions e Amazon MWAA .....	16
Coreografia .....	16
Scelta dell'approccio di coordinamento .....	17
Gestire APIs .....	18
Gateway Amazon API .....	18
Autenticazione e autorizzazione .....	18
Chiavi API e limiti di velocità .....	19
Pubblico e privato APIs .....	19
Quando usare API Gateway .....	19
Messaggistica .....	21
Amazon SQS .....	21
Polling .....	22
Linea guida .....	22
Amazon SNS .....	22
Linea guida .....	24
Amazon EventBridge .....	24
Linea guida .....	25
AWS AppSync Eventi e API Gateway .....	26
Linea guida .....	26

Domande frequenti .....	27
Come posso combinare diversi modelli di integrazione? .....	27
Qual è il vantaggio principale dell'utilizzo di un'architettura di microservizi? .....	27
Come posso implementare la gestione degli errori? .....	27
Quali sono i vantaggi del claim check pattern? .....	27
Quali sono i vantaggi del pattern di callback? .....	28
Posso implementare la comunicazione bidirezionale? .....	28
Come posso ottimizzare l'uso delle funzioni Lambda? .....	28
Quali sono le principali differenze tra l'utilizzo di Amazon SNS e? EventBridge .....	28
Risorse .....	29
Servizio AWS documentazione .....	29
Lettura aggiuntiva .....	29
Cronologia dei documenti .....	30
Glossario .....	31
# .....	31
A .....	32
B .....	35
C .....	37
D .....	40
E .....	44
F .....	46
G .....	48
H .....	49
I .....	51
L .....	53
M .....	55
O .....	59
P .....	62
Q .....	64
R .....	65
S .....	68
T .....	72
U .....	73
V .....	74
W .....	74
Z .....	75



# Integrazione di microservizi utilizzando servizi serverless AWS

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

Settembre [2025 \(storia del documento\)](#)

Un aspetto importante della modernizzazione del software dell'organizzazione è la selezione dei modelli architetturici giusti che consentano agilità e reattività alle mutevoli esigenze aziendali. In alcune applicazioni, un'architettura monolitica è una scelta comune. Tuttavia, per molte organizzazioni, il [refactoring dei monoliti in microservizi](#) può essere una strategia di modernizzazione efficace quando il caso d'uso è in linea con i vantaggi dei microservizi.

I microservizi e i monoliti non si escludono a vicenda: molte organizzazioni di successo utilizzano entrambi i modelli insieme, in cui i monoliti modulari servono alcuni domini e i microservizi gestiscono altri.

Quando i microservizi fanno parte dell'architettura, è possibile che vengano richiamati diversi servizi per recuperare i dati per una transazione commerciale. L'implementazione di queste integrazioni richiede una progettazione attenta per affrontare potenziali sfide come la coerenza dei dati, la latenza e la complessità operativa. Quando i microservizi sono integrati correttamente, possono offrire vantaggi come scalabilità indipendente, maggiore velocità di sviluppo e potenziali ottimizzazioni dei costi.

La guida fa parte di una serie di contenuti che illustra l'approccio alla modernizzazione delle applicazioni consigliato da AWS. La serie include anche:

- [Strategia per modernizzare le applicazioni in Cloud AWS](#)
- [Approccio graduale alla modernizzazione delle applicazioni in Cloud AWS](#)
- [Valutazione della preparazione alla modernizzazione per le applicazioni in Cloud AWS](#)
- [Scomposizione dei monoliti in microservizi](#)

## Destinatari principali

Questa guida è destinata ai proprietari di applicazioni, agli imprenditori, agli architetti, ai responsabili tecnici e ai project manager che hanno stabilito che i microservizi sono appropriati per il loro caso

d'uso specifico. La guida introduce diversi modelli per la comunicazione sincrona e asincrona tra microservizi utilizzando sistemi serverless come Servizi AWS Amazon API AWS Lambda Gateway per l'autonomia e la scalabilità.

## Obiettivi

Utilizzando questa guida per integrare i nuovi microservizi, è possibile trasformare in modo efficiente l'architettura della propria organizzazione in un'architettura di microservizi. Questo aiuta a fornire un rapido adattamento alle fluttuanti esigenze aziendali attraverso un'elevata scalabilità, una migliore resilienza, una distribuzione continua e l'isolamento dai guasti. Un'architettura di microservizi accelera anche l'innovazione, poiché ogni microservizio può essere implementato e testato singolarmente.

Un'architettura di microservizi può anche contribuire a ridurre i tempi di commercializzazione dei prodotti o dei servizi, poiché ogni microservizio ha una base di codice indipendente che semplifica e velocizza l'aggiunta di nuove funzionalità e l'iterazione su di esse.

## Sicurezza

È necessario proteggere adeguatamente i microservizi per proteggere l'integrità dei servizi e dei dati, ma assicurarsi che la sicurezza non influisca negativamente sulle prestazioni dell'applicazione.

In un ambiente di microservizi, è necessario considerare in che modo ciascun servizio autenticherà e autorizzerà le richieste ricevute da un client esterno o da un altro microservizio. Considerate anche come ciascun servizio accederà in modo sicuro agli altri. Servizi AWS

L'accesso a Servizi AWS deve essere concesso tramite ruoli con ambito ristretto [AWS Identity and Access Management \(IAM\)](#). Supponendo che un ruolo IAM fornisca al microservizio credenziali IAM a breve termine sotto forma di chiave di accesso, segreto di accesso e token di sessione. Questi vengono utilizzati dai vari kit di sviluppo software (SDKs) per firmare le richieste Servizi AWS utilizzando [AWS Signature Version 4 \(SigV4\)](#).

# Modelli di comunicazione

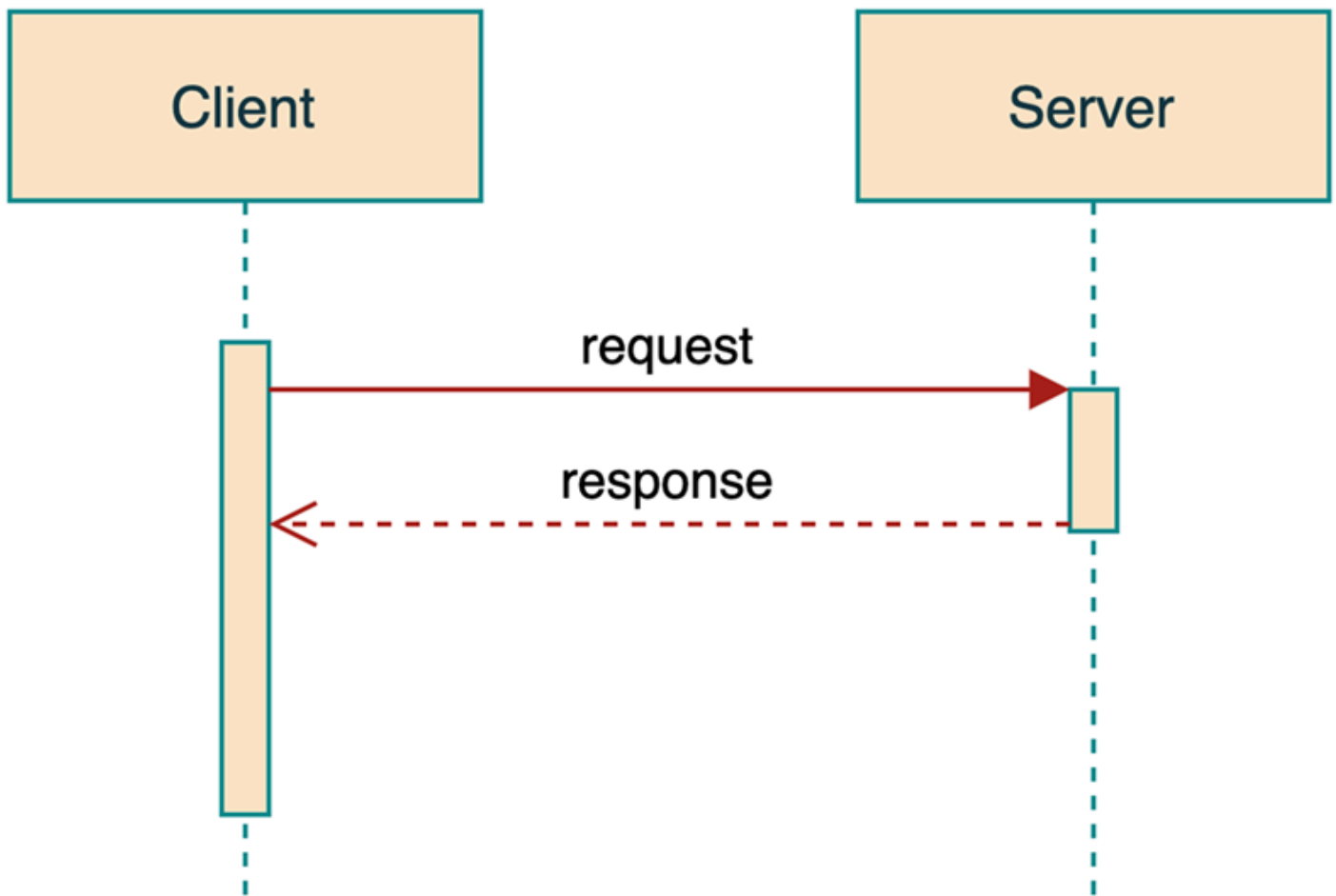
In un'architettura di microservizi, la comunicazione avviene secondo due modelli principali: sincrono e asincrono. Nella comunicazione sincrona, il chiamante attende una risposta prima di procedere, analogamente a una chiamata API HTTP REST in tempo reale. La comunicazione asincrona segue uno schema basato sui messaggi in cui il chiamante continua l'elaborazione senza attendere una risposta, ad esempio quando utilizza le code di messaggi. Le sezioni seguenti esaminano in dettaglio l'implementazione, i vantaggi e i casi d'uso di ciascun pattern.

## Argomenti

- [Comunicazione sincrona](#)
- [Comunicazione asincrona](#)

## Comunicazione sincrona

Nella comunicazione sincrona, un client invia una richiesta a un servizio, come illustrato nel diagramma seguente. Gli esempi includono una richiesta di recupero di informazioni, ad esempio una HTTP GET richiesta, o una richiesta di modifica dei dati, come una richiesta HTTP PUT. In entrambi i casi, il client attende che il server risponda prima di procedere. Le chiamate sincrone sono familiari alla maggior parte degli sviluppatori, sono facili da implementare e risolvere i problemi e, in molti casi, sono lo standard di comunicazione ampiamente accettato.



I vantaggi della comunicazione sincrona includono:

- Controllo prevedibile del flusso: esecuzione deterministica e cicli di richiesta-risposta chiari, più facili da comprendere rispetto alla comunicazione asincrona.
- Forte coerenza: conferma immediata delle modifiche ai dati e degli aggiornamenti dello stato.
- Semplice gestione degli errori: propagazione diretta di errori ed eccezioni.
- Debug semplificato: tracciamento e monitoraggio delle richieste semplici.
- Supporto dei protocolli: protocolli consolidati come HTTP e REST, che semplificano l'implementazione.

La comunicazione sincrona presenta alcuni inconvenienti:

- Accoppiamento stretto: dipendenza diretta dei servizi dalla disponibilità reciproca.
- Impatto sulla rete: aumento del carico di rete dovuto alle connessioni sempre aperte.

- Utilizzo delle risorse: maggiore utilizzo della memoria dovuto al mantenimento degli stati di connessione.
- Guasti a cascata: possibilità di propagazione rapida dei problemi di un unico servizio nel sistema.

## Comunicazione asincrona

Al contrario, nella comunicazione asincrona, il client invia una richiesta a un servizio ma non riceve una risposta immediata. In questo caso, il client di solito riceve solo una conferma dell'accettazione della richiesta.

I vantaggi della comunicazione asincrona includono:

- Supporto dell'architettura basata sugli eventi: soluzione naturale per i modelli di event-sourcing e Command Query Responsibility Segregation (CQRS).
- Migliore gestione delle risorse: capacità dei servizi di elaborare le richieste in base alla loro capacità.
- Migliore isolamento dei guasti: disaccoppiamento dei servizi, che previene i guasti a cascata.
- Gestione dei picchi di carico: migliore gestione dei picchi di traffico grazie all'accodamento dei messaggi.

Gli svantaggi includono la complessità. Esempio:

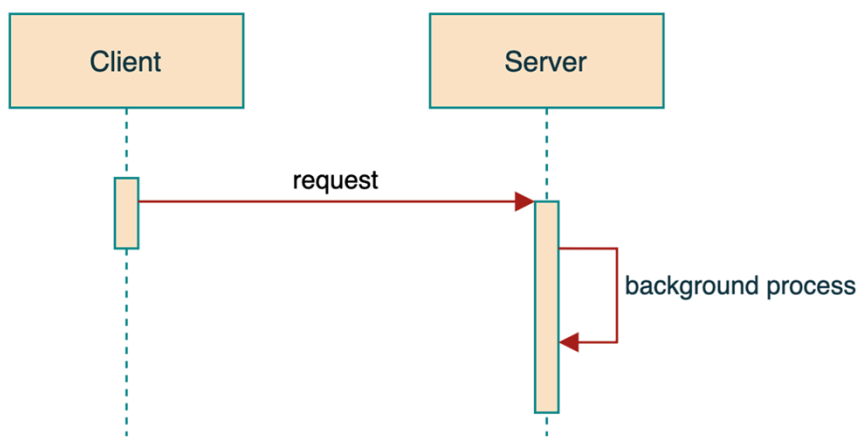
- Se il client richiede il risultato dell'operazione asincrona, l'implementazione di un meccanismo per recuperare o ricevere tale risultato richiede uno sforzo maggiore.
- La risoluzione dei problemi relativi alle operazioni asincrone può essere più difficile, poiché la risoluzione dei problemi richiede l'esame dei log su più sistemi.
- Può essere più difficile testare le operazioni asincrone, poiché il test richiede il coordinamento tra più sistemi e servizi.

Gli approcci alla comunicazione asincrona includono fire and forget, claim check, callback e comunicazione bidirezionale.

## Spara e dimentica

Nello schema fire and forget, un client invia una richiesta al server e riceve in modo sincrono una conferma che indica che il server ha ricevuto il messaggio e lo elaborerà. Tuttavia, l'elaborazione

effettiva non è ancora avvenuta e il client non ha alcuna visibilità su quando o come verrà eseguita. Il diagramma seguente illustra questo schema.



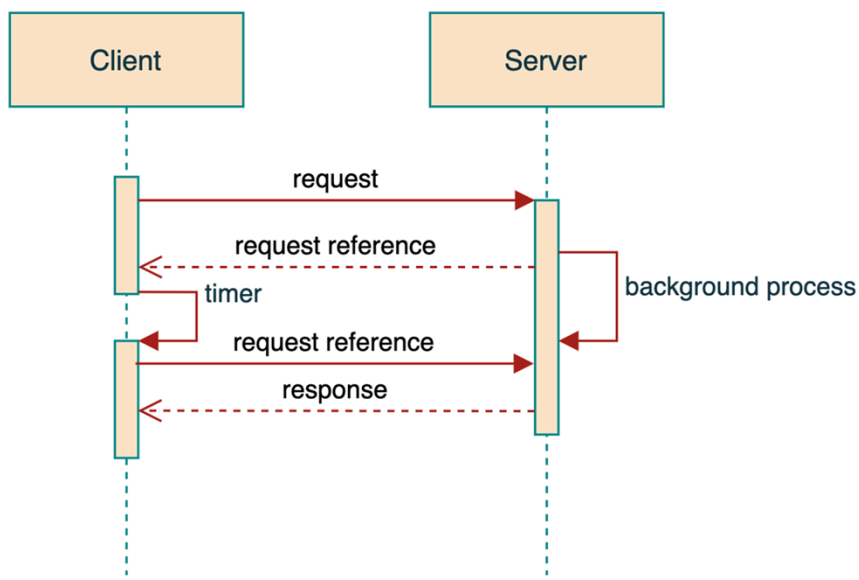
In questo caso, il servizio non deve inviare la conferma finché l'oggetto non viene mantenuto in modo duraturo. Questa persistenza può essere implementata come operazione di scrittura nel database o inserendo un elemento in una coda.

Ulteriori considerazioni:

- Implementa l'idempotenza per gestire messaggi duplicati. Cioè, ogni messaggio deve essere elaborato una sola volta.
- Prendi in considerazione [le code di lettere morte](#) in caso di elaborazione non riuscita.
- Monitora le percentuali di successo dell'elaborazione dei messaggi.

## Verifica dei reclami

Se un cliente ha bisogno del risultato di una chiamata di assistenza, puoi creare il servizio in modo che emetta un assegno di reclamo quando riceve una richiesta. Il diagramma seguente illustra questo modello. Il controllo del reclamo viene implementato come identificatore che il servizio restituisce nella sua conferma. Il client può utilizzare questo identificatore in un secondo momento per verificare lo stato della richiesta e recuperare il risultato quando la richiesta è completa.



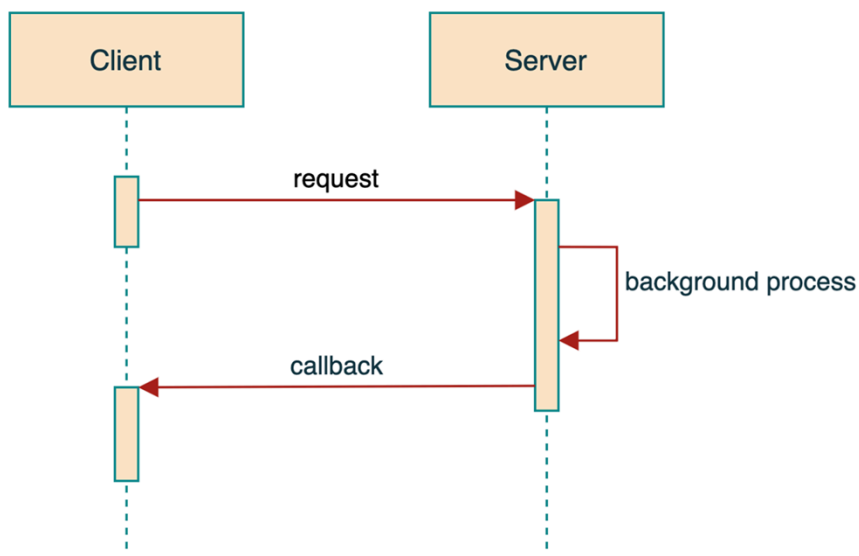
I clienti devono implementare un meccanismo per verificare i risultati. Questo può essere automatizzato (ad esempio, un controllo può essere eseguito ogni n minuti) o implementato manualmente, dove il controllo viene eseguito in risposta a un altro evento o a un'altra azione dell'utente. I servizi che implementano il modello di controllo dei reclami devono indicare esplicitamente il periodo di validità del controllo dei reclami.

Migliori pratiche:

- Implementa il backoff esponenziale per i sondaggi.
- Imposta un orario di vita appropriato (TTL) per i controlli dei reclami.
- Fornisci endpoint di stato per il monitoraggio dei progressi.

## Richiamata

Nello schema di callback, un cliente invia una richiesta a un servizio e fornisce al servizio un luogo a cui contattare il servizio una volta completata l'elaborazione. Il client non attende un risultato e l'elaborazione continua. Il servizio ha la responsabilità di contattare la sede una volta completata l'elaborazione e di fornire il risultato. I tipi più comuni di ubicazioni per le risposte sono REST APIs o code. Il diagramma seguente illustra lo schema di callback.

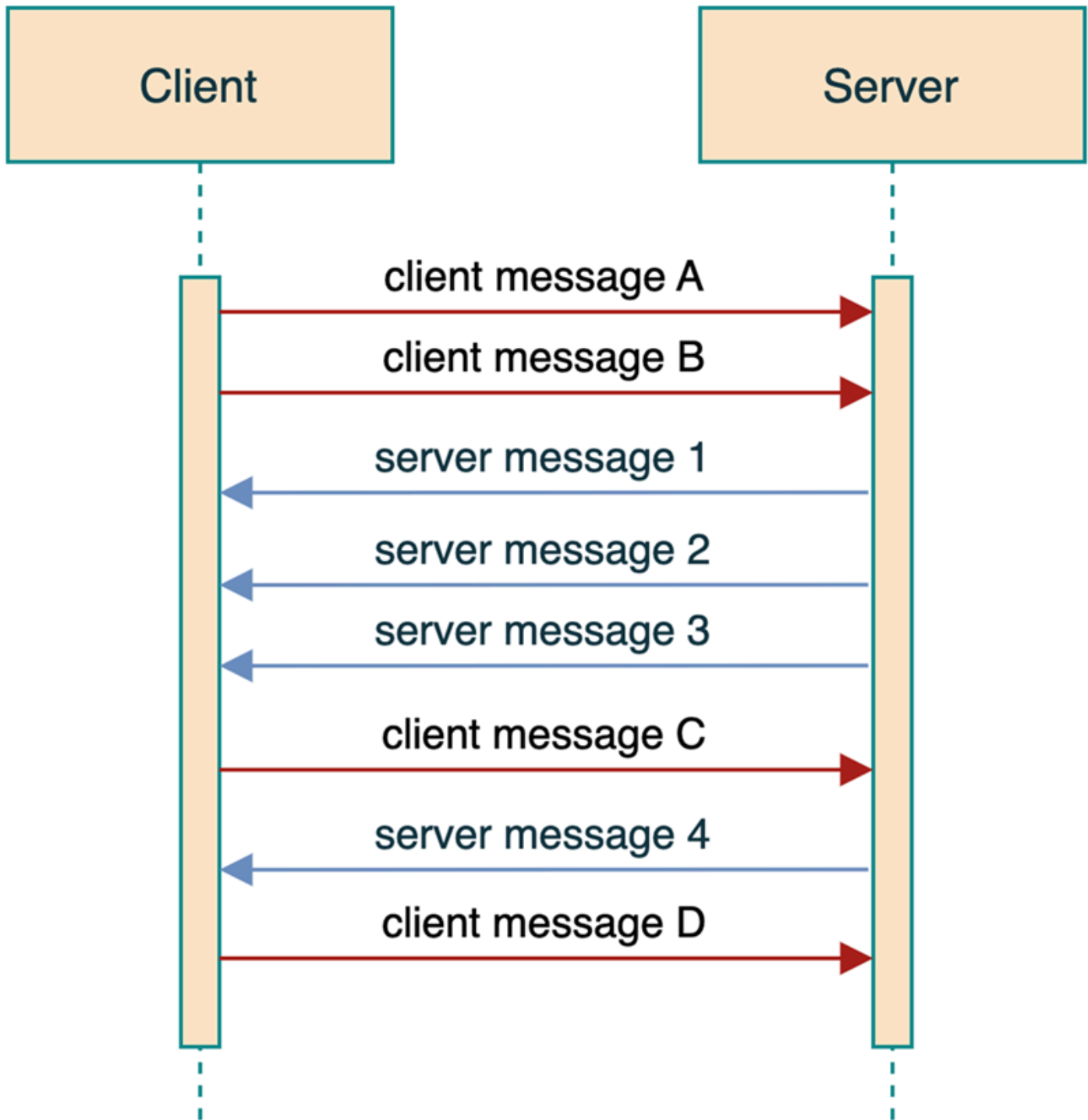


Implementazione:

- Implementa meccanismi di ripetizione per i callback falliti.
- Proteggi la posizione di callback come faresti con altri servizi.
- Gestisci i timeout di callback.

## Comunicazione bidirezionale

Per implementare la comunicazione bidirezionale, è necessario creare una connessione statica tra un client e un servizio, che consenta sia al client che al servizio di inviare ed elaborare messaggi. Il diagramma seguente ne è l'illustrazione. Sebbene la comunicazione sia asincrona, il servizio deve essere in grado di supportare una connessione aperta per ogni client.



Considerazioni sull'implementazione:

- Ordine dei messaggi
- Numeri di sequenza

- Strategie di partizione
- Ordine dei messaggi
- Gestione dello stato
  - Modelli di approvvigionamento degli eventi
  - Riconciliazione tra Stati
  - Modelli di consistenza
- Gestione degli errori
  - [Code di lettere morte](#)
  - Policy di ripetizione
  - [Interruttori automatici](#)
  - Strategie di fallback
- Monitoraggio e osservabilità
  - Correlazione IDs
  - Monitoraggio dei messaggi
  - Metriche delle prestazioni
  - Indicatori dello stato del sistema

# Opzioni di coordinamento

Le comunicazioni sincrone e asincrone funzionano bene per un client che chiama un singolo servizio o un numero limitato di servizi. Tuttavia, in un ambiente reale, questa comunicazione può diventare rapidamente complicata e difficile da scalare. La realizzazione di un'unità di lavoro potrebbe richiedere diversi microservizi, che potrebbero avere interdipendenze. Spesso, queste interazioni sono modellate come un flusso di lavoro. Esistono due approcci per progettare questi flussi di lavoro: orchestrazione e coreografia.

## Argomenti

- [Orchestrazione](#)
- [Coreografia](#)
- [Scelta dell'approccio di coordinamento](#)

## Orchestrazione

In questo approccio, un singolo orchestratore è responsabile della chiamata di ogni microservizio, della determinazione se effettuare chiamate in sequenza o in parallelo, della manipolazione delle risposte dei singoli servizi lungo il percorso e della compilazione del risultato finale. Un orchestratore può combinare chiamate sincrone e asincrone.

[AWS Step Function](#) e [Amazon Managed Workflows for Apache Airflow \(Amazon MWAA\)](#) sono ottime scelte per gli orchestratori di flussi di lavoro.

L'orchestrazione è una buona scelta quando ci sono rami logici nel processo e hai bisogno di un unico posto per incapsulare tale logica. È utile anche quando si desidera implementare il modello asincrono di controllo dei reclami. Ad esempio, i flussi di lavoro standard in Step Functions possono mettere in pausa un flusso di lavoro e attendere una richiamata da un altro servizio. L'uso di un orchestratore migliora anche il monitoraggio e l'osservabilità di un processo.

## Esempio: Step Functions

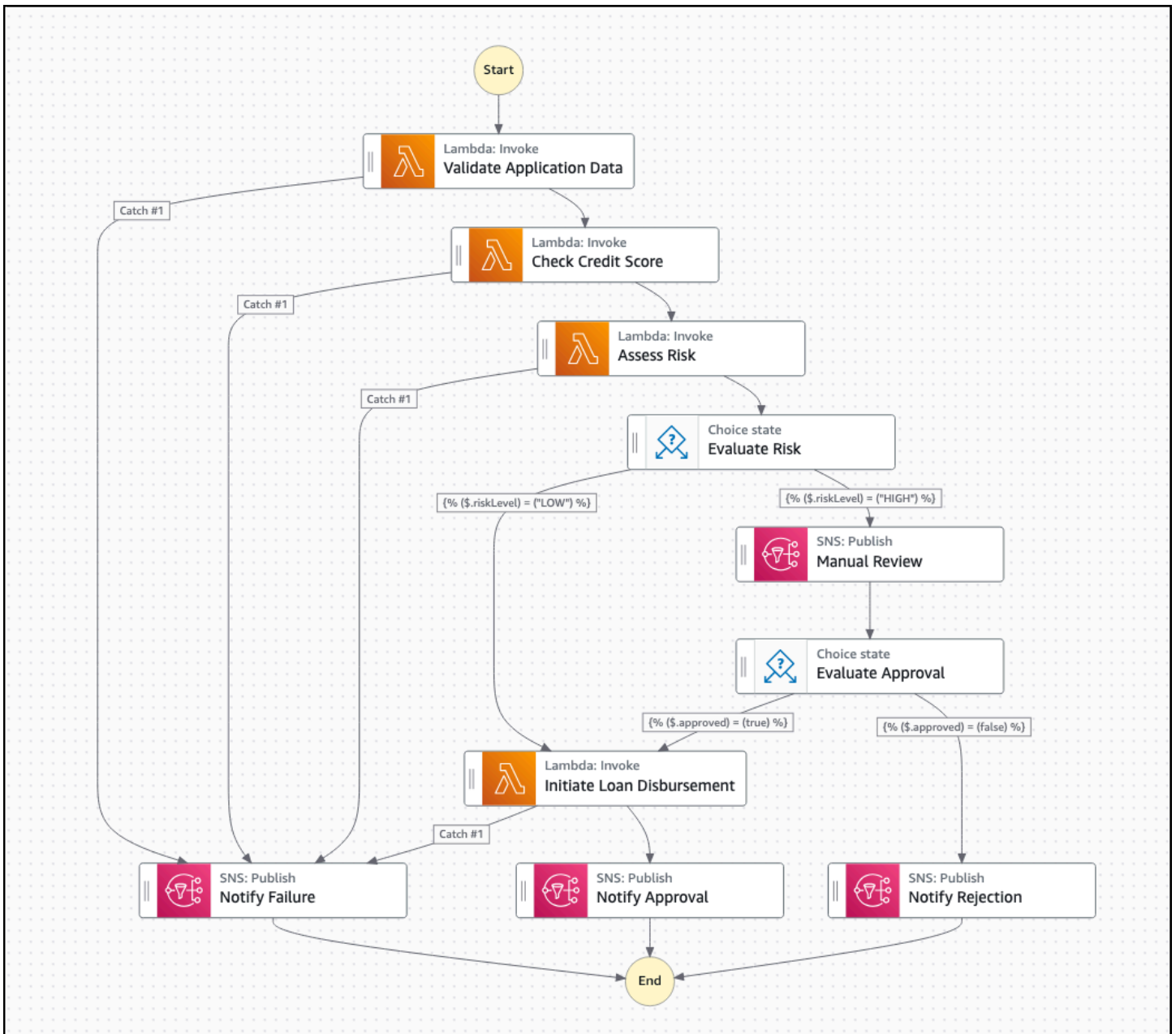
Puoi usare Step Functions per coordinare più funzioni Lambda e altre Servizi AWS, per creare flussi di lavoro complessi per l'integrazione di microservizi. Questa opzione è particolarmente utile per processi di lunga durata e in più fasi che coinvolgono diversi microservizi.

Dovresti prendere in considerazione l'utilizzo di Step Functions se:

- L'integrazione dei microservizi implica processi complessi e in più fasi.
- È necessario mantenere lo stato durante le operazioni di lunga durata.
- Desiderate implementare la gestione degli errori e la logica dei tentativi a livello di flusso di lavoro.
- È necessario coordinare le operazioni sincrone e asincrone.

Step Functions offre un editor visivo per la progettazione di flussi di lavoro complessi, che semplifica il processo di creazione e gestione di macchine a stati. Fornisce meccanismi integrati di gestione degli errori, tra cui la logica dei tentativi e la gestione dello stato di errore, che migliorano l'affidabilità e la robustezza delle applicazioni. I flussi di lavoro standard supportano processi di lunga durata fino a un anno, il che è adatto per flussi di lavoro che si estendono per periodi prolungati. Questa opzione separa la logica di orchestrazione dal codice dell'applicazione, in modo da ridurre significativamente la complessità del codice. Ciò significa che gli sviluppatori possono concentrarsi sulla logica aziendale principale mentre Step Functions gestisce il controllo del flusso e il coordinamento dei componenti distribuiti.

Ad esempio, si consideri un processo di approvazione di un prestito in un'applicazione di servizi finanziari, illustrato nel diagramma seguente. Il processo inizia quando viene presentata una richiesta di prestito.



Nella macchina a stati illustrata nel diagramma precedente, Step Functions orchestra i seguenti passaggi:

- Convalida dei dati dell'applicazione (funzione Lambda)
- Verifica il punteggio di credito (funzione Lambda che richiama un'API esterna)
- Valutazione del rischio (funzione Lambda)
- Se il rischio è elevato, passa alla revisione manuale (attività di approvazione umana)
- Se approvato, avvia l'erogazione del prestito (funzione Lambda)

- Invia una notifica al richiedente (Amazon SNS)

Puoi utilizzare questo approccio per gestire in modo affidabile un processo complesso e potenzialmente di lunga durata, con una gestione degli errori integrata e la possibilità di includere passaggi automatici e manuali.

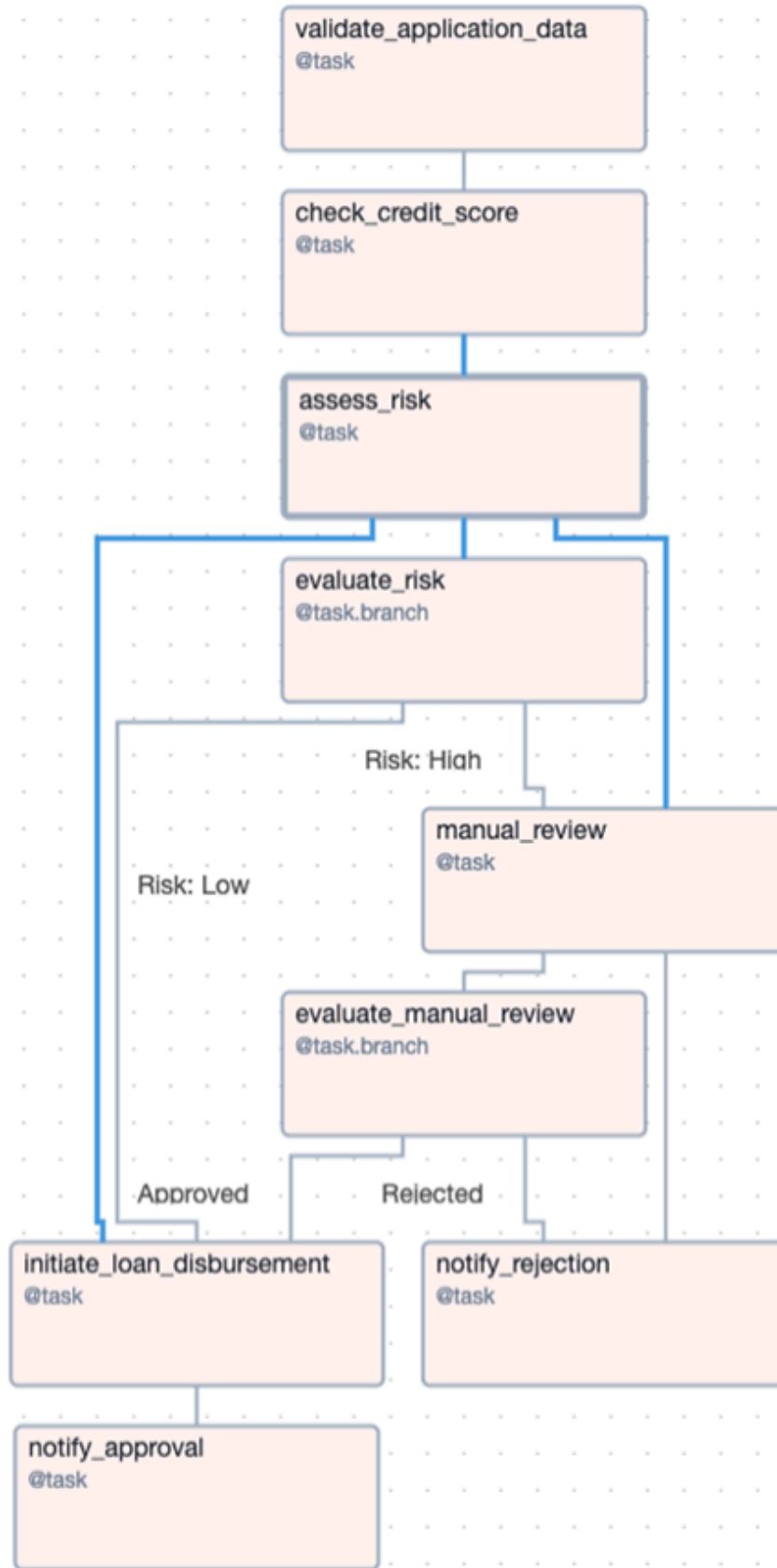
Considerazioni:

- Progetta attentamente la tua macchina a stati per gestire tutti gli scenari possibili.
- Esegui i passaggi in parallelo ove possibile.
- Utilizza i meccanismi integrati di gestione degli errori e riprova in Step Functions per errori permanenti e temporanei.
- Prendi in considerazione l'utilizzo di [flussi di lavoro standard o rapidi](#) in base al tuo caso d'uso. I flussi di lavoro rapidi potrebbero essere preferibili per flussi di lavoro di breve durata o ad alto volume.
- [Monitora le metriche di esecuzione](#) per ottimizzare il flusso di lavoro.
- Utilizza flussi di lavoro annidati per incapsulare e riutilizzare le funzionalità su più macchine a stati.
- Per flussi di lavoro complessi, prendi in considerazione l'utilizzo di [Amazon Bedrock Agents](#) come alternativa a Step Functions.

Per ulteriori informazioni, consulta la [documentazione di Step Functions](#).

## Esempio: Amazon MWAA

Se la tua organizzazione utilizza già Apache Airflow, Amazon MWAA è una scelta naturale come orchestratore di flussi di lavoro. In Apache Airflow, puoi creare flussi di lavoro come grafici aciclici diretti () usando Python. DAGs La rappresentazione DAG della macchina a stati illustrata nella sezione Step Functions potrebbe avere il seguente aspetto:



Per informazioni su come lavorare DAGs, consulta la documentazione di [Amazon MWAA](#).

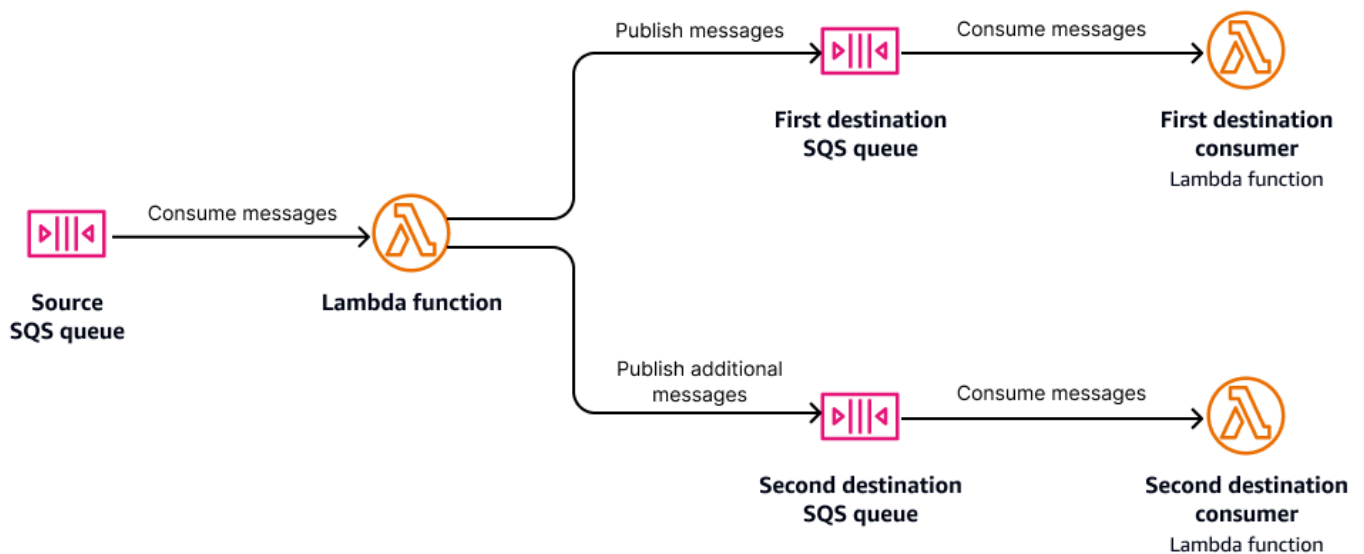
## Principali differenze tra Step Functions e Amazon MWAA

- Step Functions è un servizio serverless completamente gestito, quindi non è necessario predisporre alcuna infrastruttura e non è necessario pianificare una finestra di manutenzione. Amazon MWAA deve essere distribuito in anticipo e sei tu a scegliere la dimensione e il numero di nodi nel cluster.
- In Step Functions, è possibile creare macchine a stati in vari modi, tra cui Workflow Studio, direttamente come JSON o utilizzando AWS Cloud Development Kit (AWS CDK) Apache DAGs Airflow è scritto in Python.
- Con Step Functions, non è previsto alcun costo in assenza di flussi di lavoro in esecuzione. Con Amazon MWAA, dovrai sostenere dei costi anche quando non è in esecuzione. DAGs

## Coreografia

In un sistema coreografato, i singoli componenti ricevono un compito, eseguono alcuni lavori ed eventualmente emettono un compito per il lavoro successivo. Non esiste un meccanismo di orchestrazione centrale. La coreografia semplifica la scalabilità dei servizi in modo indipendente, poiché ogni servizio opera in relativo isolamento. Esegue il lavoro quando lo riceve, indipendentemente dalla velocità di trasmissione di cui è capace il servizio. La coreografia è spesso una parte centrale di un'architettura [basata sugli eventi \(EDA\)](#).

Nel diagramma seguente, non esiste alcuna coordinazione tra le funzioni Lambda. Ogni funzione elabora solo i messaggi nella coda sottoscritta. Ogni funzione è responsabile della propria gestione degli errori e può controllare la concorrenza, ad esempio se una dipendenza a valle ha un limite di richieste al secondo (RPS).



Un'EDA offre una serie di vantaggi, come l'abbinamento libero dei servizi e l'estensibilità. Una discussione completa dei principi dell'EDA non rientra nell'ambito di questa guida. Per ulteriori informazioni, consulta:

- [AWS Well-Architected Framework: lente applicativa senza server](#)
- [Introduzione all'architettura basata sugli eventi \(Serverless Land\)](#)
- [Passaggio all'architettura basata sugli eventi \(Serverless Developer Guide\)](#)

## Scelta dell'approccio di coordinamento

Sia la coreografia che l'orchestrazione hanno i loro usi nell'integrazione dei microservizi. Scegliete la coreografia entro i confini di un singolo microservizio, dove avete il pieno controllo sulle dipendenze. Scegliete l'orchestrazione quando lavorate oltre i confini dei microservizi. Ad esempio, più microservizi che partecipano a una transazione distribuita trarranno vantaggio dall'orchestrazione per tenere conto del ripristino in caso di guasti. I microservizi che gestiscono eventi che potrebbero interessare altri microservizi trarranno vantaggio dalla coreografia e da un'architettura basata sugli eventi.

Uno schema comune per implementare il rollback quando più sistemi sono coinvolti in una singola transazione è il modello saga.

# Gestire APIs

Una corretta gestione delle API rende i microservizi accessibili ai consumatori interni ed esterni. AWS offre una varietà di servizi che potete utilizzare insieme per esporre in modo sicuro il vostro microservizio. Questi servizi consentono di rafforzare la sicurezza e di implementare il monitoraggio API e l'osservabilità da una posizione centrale. Puoi anche utilizzare [Amazon CloudFront](#) per migliorare le prestazioni se i tuoi utenti API sono geograficamente distanti dal luogo in Regione AWS cui sono ospitati i servizi.

## Gateway Amazon API

[Amazon API Gateway](#) è un servizio completamente gestito che consente agli sviluppatori di creare, pubblicare, mantenere, monitorare e proteggere REST WebSocket APIs su qualsiasi scala. Puoi utilizzare API Gateway per implementare molti dei modelli descritti nella sezione [Modelli di comunicazione](#) di questa guida.

Esistono due tipi principali di REST APIs: REST e HTTP. Entrambi i tipi supportano RESTful APIs ma offrono funzionalità diverse. Per determinare quale si adatta meglio alle tue esigenze, consulta [Scegli tra REST APIs e HTTP APIs](#) nella documentazione di API Gateway. Questa sezione della guida si concentra su API Gateway REST APIs.

L'utilizzo di API Gateway come punto di ingresso APIs fornisce un unico luogo in cui implementare problemi comuni, come la convalida e la sicurezza delle richieste. API Gateway REST APIs offre [la convalida delle richieste](#), che consente di definire il formato delle richieste utilizzando lo schema [JSON](#). API Gateway convalida le richieste in entrata rispetto allo schema definito e rifiuta le richieste con formato errato.

## Autenticazione e autorizzazione

API Gateway REST APIs supporta i seguenti meccanismi di autenticazione (AuthN) e autorizzazione (AuthZ):

- IAM: se utilizzi IAM, le richieste all'API devono essere firmate utilizzando [AWS Signature Version 4 \(SigV4\)](#).
- Amazon Cognito — API Gateway verificherà che un token al portatore presentato sia stato emesso da un pool di utenti di Amazon Cognito. Puoi anche configurare il tuo pool di utenti Amazon Cognito per l'integrazione con un provider di identità (IdP) di terze parti, se ne stai già utilizzando uno. Puoi anche utilizzare un pool di utenti Amazon Cognito per l'autenticazione machine-to-machine (M2M).

- AWS Lambda authorizer: API Gateway invocherà una funzione Lambda specificata per eseguire i controlli desiderati, per determinare se una richiesta deve essere autorizzata.

Per ulteriori informazioni, consulta [Controllare e gestire l'accesso a REST APIs](#) nella documentazione di API Gateway.

## Chiavi API e limiti di velocità

Puoi controllare chi è autorizzato a chiamarti APIs e con quale frequenza utilizzando le chiavi API e i piani di utilizzo. Le chiavi API non devono essere utilizzate per l'autenticazione, ma possono essere utilizzate insieme agli schemi menzionati in precedenza. Gli utenti non devono sempre fornire la propria chiave API: ad esempio, gli autorizzatori Lambda possono restituire una chiave API per un utente. Il piano di utilizzo consente di specificare il throughput, il limite di burst e la quota mensile. Per ulteriori informazioni, consulta [Piani di utilizzo e chiavi API per REST APIs](#) nella documentazione di API Gateway.

## Pubblico e privato APIs

API Gateway REST APIs accessibili tramite Internet supporta due tipi di endpoint:

- Ottimizzato per Edge, il che significa che le richieste dei chiamanti vengono indirizzate a un CloudFront punto di presenza (POP) nelle vicinanze. Ciò può comportare un miglioramento delle prestazioni per clienti geograficamente diversi.
- Regionale, il che significa che le richieste vengono indirizzate a una risorsa all'interno di una determinata risorsa. Regione AWS Questa è una buona scelta quando tutti i tuoi clienti si trovano vicini alla regione in cui è distribuita l'API.

API Gateway REST supporta APIs anche endpoint API privati, accessibili da un cloud privato virtuale (VPC) utilizzando un endpoint VPC di interfaccia. Puoi anche condividere in modo sicuro il REST privato APIs creando endpoint VPC di interfaccia in VPCs altri e anche altri. Account AWS Per ulteriori informazioni, consulta la sezione [Tipi di endpoint API per REST APIs](#) nella documentazione di API Gateway.

## Quando usare API Gateway

API Gateway è un'ottima scelta per i servizi RESTful Web e WebSocket le connessioni in tempo reale. Quando si utilizza WebSocket APIs in API Gateway, è possibile aggiungere comportamenti

per gli eventi di connessione e disconnessione, ad esempio l'archiviazione della connessione IDs in un data store esterno associato agli attributi del client. Puoi anche indirizzare le richieste a comportamenti personalizzati utilizzando gli attributi dei messaggi.

Sia REST che WebSocket APIs possono integrarsi direttamente con molti altri Servizi AWS senza richiedere risorse di calcolo separate come le funzioni Lambda. Ciò può migliorare le prestazioni e ridurre i costi.

REST APIs supporta sia il routing basato sul percorso che quello basato sulle intestazioni e puoi utilizzarli separatamente o insieme. Uno schema comune consiste nel fornire un'API REST come porta d'ingresso per una serie di problemi condivisi APIs, come discusso in precedenza, e quindi comportarsi come un proxy inverso e indirizzare le richieste autorizzate all'endpoint API corretto.

# Messaggistica

Come illustrato nella sezione [Schemi di comunicazione](#), è possibile utilizzare la messaggistica per comunicare in modo sincrono o asincrono tra i servizi. Esistono molti servizi AWS serverless tra cui scegliere e la scelta deve essere basata sulle esigenze di integrazione. Ad esempio, se richiedi una consegna ordinata dei messaggi, dovresti scegliere un servizio come Amazon Simple Queue Service (Amazon SQS) o Amazon Simple Notification Service (Amazon SNS). Entrambi i servizi supportano la distribuzione First In First Out (FIFO), a differenza di Amazon EventBridge, che non la supporta.

Le sezioni seguenti illustrano questi servizi in modo più dettagliato.

## Argomenti

- [Amazon SQS](#)
- [Amazon SNS](#)
- [Amazon EventBridge](#)
- [AWS AppSync Eventi e API Gateway](#)

## Amazon SQS

[Amazon SQS](#) supporta code standard, che non garantiscono l'ordine, e code FIFO, che garantiscono l'ordinamento all'interno di un determinato gruppo di messaggi.

Le code sono un metodo comune per coreografare microservizi e forniscono uno spazio di archiviazione durevole per i messaggi per un massimo di 14 giorni. Le code sono occupate dai produttori e esaurite dai consumatori. Quando si utilizza AWS Lambda come consumatore, è possibile configurare una coda SQS come origine di eventi. In questo caso, il servizio Lambda Service Event Source Mapping (ESM) esegue il polling della coda per te e consegna i messaggi alla funzione Lambda non appena diventano disponibili. I microservizi che funzionano su altri tipi di servizi di elaborazione, come Amazon Elastic Container Service (Amazon ECS) o Amazon Elastic Compute Cloud (EC2Amazon), devono implementare un proprio meccanismo di polling per recuperare nuovi messaggi dalla coda, quando diventano disponibili.

Lambda ESM per Amazon SQS supporta anche il filtraggio dei messaggi, che consente di elaborare solo un sottoinsieme di messaggi in una coda in base al contenuto del corpo del messaggio.

## Polling

Amazon SQS supporta il polling breve e il polling lungo dei messaggi. Short polling interroga un sottoinsieme di server per trovare i messaggi disponibili e li restituisce immediatamente. Tuttavia, potrebbe non restituire tutti i messaggi disponibili. Ciò è utile quando l'applicazione deve consumare i messaggi il più rapidamente possibile o non può tollerare un'attesa per un periodo di tempo più lungo.

Il polling attende a lungo che sia trascorso un periodo di tempo configurabile o che venga ricevuto un numero configurabile di messaggi prima di restituire i messaggi. Ciò potrebbe ridurre il numero di sondaggi vuoti, ovvero il numero di sondaggi in cui non viene restituito alcun messaggio, in particolare per le code che non ricevono molti messaggi. La riduzione del numero di sondaggi vuoti può ridurre i costi di Amazon SQS, poiché questo servizio addebita un costo per ogni richiesta e ogni operazione di polling è una richiesta.

## Linea guida

Le code sono un'ottima scelta quando:

- Desiderate disaccoppiare i componenti e non è necessaria una comunicazione sincrona tra di essi.
- Si sta comunicando tra componenti che hanno accordi sui livelli di servizio (SLAs) o obiettivi di livello di servizio () diversi di disponibilità. SLOs
- In genere si dispone di un unico consumatore per una serie di messaggi.

Prendi in considerazione un'opzione alternativa se:

- È necessaria una comunicazione sincrona.
- È necessaria una logica di routing complicata per inviare messaggi al consumatore corretto.

## Amazon SNS

[Amazon SNS](#) ti consente di creare argomenti standard e FIFO. Gli argomenti vengono utilizzati per implementare un'architettura publish/subscribe (pub/sub). Amazon SNS supporta diversi tipi di abbonamento, tra cui e-mail, SMS (supponendo che tu abbia configurato un'identità di origine, come un numero verde o un codice lungo di 10 cifre), endpoint HTTP (S) e code SQS. Le sottoscrizioni degli utenti finali, come messaggi e-mail e SMS, a un argomento SNS devono essere confermate dall'abbonato. Amazon SNS consente un'ampia diffusione dei servizi, il che significa che un singolo

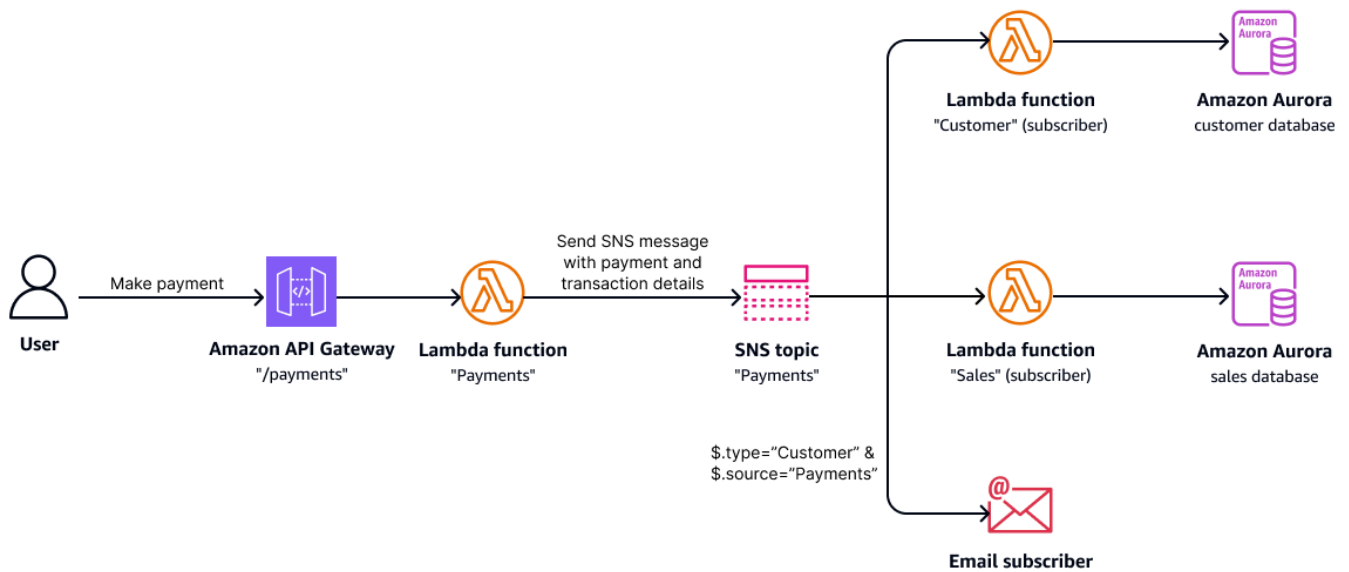
messaggio può essere recapitato a un numero potenzialmente elevato di abbonati. Un argomento standard SNS ha un limite predefinito di 12,5 milioni di abbonamenti.

In un ambiente di microservizi, un argomento SNS è utile per separare il routing dei messaggi e la logica di recapito da un editore. Questo può essere implementato utilizzando filtri per argomento. Concettualmente, i filtri per argomento sono in qualche modo simili EventBridge alle regole di Amazon, ma sono configurati per ogni abbonato anziché essere disponibili da una posizione centralizzata. Ad esempio, supponiamo che tu abbia:

- Un servizio di gestione degli ordini, che elabora gli ordini.
- Un servizio di evasione ordini, che gestisce l'evasione degli ordini.
- Un servizio di fidelizzazione, che assegna punti fedeltà ai membri per gli ordini.

Quando un ordine è pronto per essere evaso, pubblica un messaggio su un argomento. Il servizio Fulfillment sottoscrive l'argomento ma non applica un filtro, perché desidera conoscere tutti gli ordini. Immagina di avere un servizio di fidelizzazione, che si occupa dell'assegnazione di punti ai membri quando effettuano un ordine. Tuttavia, non tutti gli ordini vengono effettuati dai membri. Il servizio di fidelizzazione sottoscriverebbe l'argomento ma implementerebbe un filtro di iscrizione per controllare un attributo che indica se l'ordine era per un membro o un ospite.

Si consideri il caso in cui un sistema riceve una richiesta da un utente finale di effettuare un pagamento, come illustrato nel diagramma seguente. In questo caso, è necessario che più sistemi a valle sappiano che la richiesta è stata effettuata in modo da poter intraprendere varie azioni. Quando usi Amazon SNS, i pagamenti vengono pubblicati su un argomento SNS e le funzioni Lambda si iscrivono all'argomento per aggiornare i database dei clienti e delle vendite. Inoltre, un abbonamento e-mail (che deve essere confermato dal cliente) invia un'e-mail di conferma al cliente utilizzando un filtro di abbonamento.



## Linea guida

Alcune delle funzionalità descritte in questa sezione per Amazon SNS si sovrappongono alle funzionalità offerte da un bus di eventi, ad esempio. EventBridge Prendi in considerazione l'utilizzo di Amazon SNS quando:

- Avrai un gran numero di abbonati a un argomento.
- Desideri utilizzare tipi di abbonamento (come e-mail o SMS), che non sono supportati nativamente da EventBridge
- Gli abbonati dovrebbero essere in grado di determinare il loro filtro di abbonamento.
- È necessario un recapito ordinato agli abbonati (per gruppo di messaggi).

Se hai molti argomenti e vengono utilizzati abbonamenti e filtri per instradare i messaggi tra i microservizi, EventBridge è probabilmente la scelta migliore.

## Amazon EventBridge

[Amazon EventBridge](#) è un servizio di bus eventi senza server e spesso funge da base per un'architettura basata sugli eventi (EDA). Puoi anche usarlo per instradare e recapitare messaggi in modo asincrono tra microservizi. Utilizzando EventBridge, i produttori pubblicano gli eventi su un autobus. Configurate le regole che corrispondono agli eventi in base al loro contenuto e scegliete uno

o più obiettivi a cui verranno consegnati gli eventi che corrispondono a quella regola. EventBridge supporta un'[ampia gamma di obiettivi di regole](#). L'utilizzo di un bus di eventi consente di separare i produttori dai consumatori e di consolidare la logica di routing e distribuzione.

In EventBridge, puoi anche creare regole pianificate, in modo da poter intraprendere azioni in momenti specifici. È possibile definire gli eventi utilizzando espressioni basate su cron e sulla frequenza.

[EventBridgePipes](#) consente di reindirizzare i messaggi da un'[origine](#) a una [destinazione](#) senza utilizzare un servizio di elaborazione come AWS Lambda. Ad esempio, supponiamo di avere una coda SQS che riceve messaggi, che dovrebbe attivare una AWS Step Functions macchina a stati. Invece di creare una funzione Lambda con una mappatura dell'origine degli eventi per consumare i messaggi dalla coda e scrivere codice utilizzando un AWS SDK per richiamare la macchina a stati, puoi usare EventBridge Pipes per farlo al posto tuo senza scrivere alcun codice personalizzato.

EventBridge è comunemente usato con altri servizi di messaggistica come Amazon SQS e Amazon SNS. Ad esempio, l'invio di eventi a una coda SQS offre a un servizio di ricezione la flessibilità di utilizzare i messaggi quando possibile, a una velocità indipendente dalla velocità di produzione degli eventi. Allo stesso modo, puoi fornire eventi che dovrebbero essere diffusi a un gran numero di abbonati a un argomento SNS.

## Linea guida

Da utilizzare quando: EventBridge

- Non è necessaria la comunicazione sincrona tra i servizi.
- Vuoi disaccoppiare la logica di routing dei messaggi dai tuoi microservizi. I microservizi producono solo eventi e li pubblicano sul bus degli eventi, mentre i servizi interessati creano regole per abbinare e fornire tali eventi.
- È necessario recapitare messaggi da un servizio supportato a un altro.

Prendi in considerazione altri servizi in situazioni in cui:

- È necessario un ordine rigoroso degli eventi. In questi casi, prendi in considerazione le code FIFO di Amazon SQS o gli argomenti FIFO di Amazon SNS. In alternativa, prendi in considerazione servizi di streaming di eventi come Amazon Kinesis Data Streams o Amazon Managed Streaming for Apache Kafka (Amazon MSK).

# AWS AppSync Eventi e API Gateway

AWS AppSync Events e Amazon API Gateway offrono entrambi un' WebSocket esperienza gestita per i tuoi microservizi.

[AWS AppSync Events](#) offre un'esperienza semplificata per la messaggistica in tempo reale utilizzando un. WebSocket AWS AppSync Events supporta la messaggistica unicast e multicast e un raggruppamento flessibile di canali in namespace, con supporto per i caratteri jolly. I microservizi possono comunicare tra loro in diversi modi utilizzando Events. AWS AppSync Ad esempio, un servizio che riceve dati in tempo reale può trasformare e pubblicare i dati sul canale appropriato, dove gli abbonati li riceveranno in tempo reale.

Inoltre, [API Gateway](#) supporta WebSocket APIs. Puoi definire integrazioni con Servizi AWS, ad esempio, Amazon DynamoDB AWS Lambda e configurare espressioni di selezione del percorso mappate su tali integrazioni. API Gateway dispone di percorsi speciali che puoi utilizzare per autorizzare e gestire le tue WebSocket connessioni. A seconda delle esigenze, è possibile archiviare le informazioni di WebSocket connessione in un data store come DynamoDB. Utilizzando queste informazioni, i messaggi possono essere pubblicati su WebSocket connessioni specifiche tramite un'API REST, con un ID di connessione specifico.

## Linea guida

Usa AWS AppSync Events quando:

- Disponi di più canali di messaggistica raggruppati in namespace e desideri pubblicare e sottoscrivere gruppi di canali utilizzando i caratteri jolly.
- Le comunicazioni avvengono principalmente tra sistemi diversi anziché tra sistemi diversi. Servizi AWS

Utilizza API Gateway WebSocket APIs quando:

- Vuoi consentire ai clienti di avere connessioni persistenti in tempo reale alle Servizio AWS integrazioni.
- Vuoi gestire le WebSocket connessioni da solo. Ad esempio, potresti voler consentire ad altri sistemi di inviare messaggi a un determinato client dopo aver cercato il relativo ID di connessione.
- Desideri utilizzare le funzionalità di API Gateway come le distribuzioni in fasi o le integrazioni proxy oppure desideri configurare i tuoi sottoprotocolli.

## Domande frequenti

### Come posso combinare diversi modelli di integrazione?

Nella maggior parte delle situazioni, ti consigliamo di combinare modelli di integrazione. Ad esempio, è possibile AWS Step Functions orchestrare un processo che richiama un servizio remoto utilizzando il claim check pattern. Oppure, potresti utilizzare un processo orchestrato che mette i messaggi in code, che a loro volta attivano servizi coreografici.

### Qual è il vantaggio principale dell'utilizzo di un'architettura di microservizi?

I principali vantaggi includono la scalabilità indipendente dei servizi, un migliore isolamento degli errori, una maggiore velocità di sviluppo grazie al lavoro in team parallelo e la capacità di distribuzione e implementazione continue (CI/CD).

### Come posso implementare la gestione degli errori in questi modelli?

È possibile implementare la gestione degli errori utilizzando i meccanismi integrati in Servizi AWS. Ad esempio, AWS Lambda le funzioni possono essere configurate con la logica di ripetizione e Amazon SQS supporta code di lettere morte per la gestione degli errori persistenti. Inoltre, Step Functions fornisce meccanismi di gestione degli errori e di nuovi tentativi a livello di flusso di lavoro.

### Quali sono i vantaggi dell'utilizzo del modello di controllo dei reclami nella comunicazione asincrona?

Lo schema di verifica del reclamo consente ai clienti di ricevere un identificatore al momento dell'invio della richiesta. Questo identificatore può essere utilizzato in seguito per controllare lo stato e recuperare il risultato. Questo modello offre vantaggi ai clienti in quanto fornisce un meccanismo per verificare i risultati senza attendere in modo sincrono. Per ulteriori informazioni, consulta la sezione [Verifica dei reclami](#) precedente di questa guida.

## In che modo il pattern di callback migliora la comunicazione asincrona nei microservizi?

Il pattern di callback migliora la comunicazione asincrona consentendo al client di fornire un punto di contatto per il servizio al termine dell'elaborazione. Ciò dissocia il client dall'attesa di una risposta e gli consente di continuare con altre attività. Per ulteriori informazioni, consulta la sezione [Callback](#) precedente di questa guida.

## Posso implementare la comunicazione bidirezionale nei microservizi utilizzando i modelli descritti?

È possibile implementare la comunicazione bidirezionale creando una connessione statica tra un client e un servizio, in modo che possano inviare ed elaborare i messaggi in modo asincrono. Ciò richiede che il servizio supporti una connessione aperta per ogni client. Per ulteriori informazioni, consultate la sezione [Comunicazione bidirezionale riportata](#) all'inizio di questa guida.

## Come posso ottimizzare l'uso delle funzioni Lambda nei modelli di comunicazione asincroni?

Puoi ottimizzare le funzioni Lambda assicurandoti che siano idempotenti per gestire potenziali duplicazioni di messaggi, utilizzando le funzionalità di Amazon SQS come i gruppi di messaggi per gli ordini e implementando polling lunghi per ridurre i costi. Inoltre, puoi monitorare i parametri di esecuzione per identificare opportunità di ottimizzazione.

## Quali sono le differenze principali tra l'utilizzo di Amazon SNS e EventBridge per il pub/sub pattern?

Amazon SNS invia un unico messaggio a tutti gli abbonati, che può includere dati non necessari per alcuni abbonati. Amazon EventBridge consente un controllo più granulare consentendoti di avere diverse regole che corrispondono a un singolo evento, ognuna delle quali attiva un servizio o un'azione downstream diversi. Per ulteriori informazioni, consulta [Amazon SNS e le EventBridge](#) sezioni precedenti di questa guida.

# Risorse

## Servizio AWS documentazione

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

## Lettura aggiuntiva

- [Strategia per modernizzare le applicazioni in Cloud AWS](#)
- [Approccio graduale alla modernizzazione delle applicazioni in Cloud AWS](#)
- [Valutazione della preparazione alla modernizzazione per le applicazioni in Cloud AWS](#)
- [Scomposizione dei monoliti in microservizi](#)
- [Implementazione di modelli di integrazione aziendale con i servizi di messaggistica: canali AWS point-to-point](#)
- [Messaggistica Pub/sub: notifiche asincrone di eventi](#)

## Cronologia dei documenti

La tabella seguente descrive le modifiche significative apportate a questa guida. Per ricevere notifiche sugli aggiornamenti futuri, puoi abbonarti a un [feed RSS](#).

Modifica	Descrizione	Data
<a href="#">Aggiornamenti principali</a>	Guida ampliata, riorganizzata e aggiornata per riflettere Servizio AWS gli aggiornamenti.	10 settembre 2025
<a href="#">Pubblicazione iniziale</a>	—	11 gennaio 2021

# AWS Glossario delle linee guida prescrittive

I seguenti sono termini di uso comune nelle strategie, nelle guide e nei modelli forniti da AWS Prescriptive Guidance. Per suggerire voci, utilizza il link [Fornisci feedback](#) alla fine del glossario.

## Numeri

### 7 R

Sette strategie di migrazione comuni per trasferire le applicazioni sul cloud. Queste strategie si basano sulle 5 R identificate da Gartner nel 2011 e sono le seguenti:

- **Refactor/re-architect** — Sposta un'applicazione e modificala sfruttando appieno le funzionalità native del cloud per migliorare l'agilità, le prestazioni e la scalabilità. Ciò comporta in genere la portabilità del sistema operativo e del database. Esempio: migra il tuo database Oracle locale all'edizione Amazon PostgreSQL-Compatible Aurora.
- **Ridefinire la piattaforma (lift and reshape)**: trasferisci un'applicazione nel cloud e introduci un certo livello di ottimizzazione per sfruttare le funzionalità del cloud. Esempio: migra il tuo database Oracle locale ad Amazon Relational Database Service (Amazon RDS) per Oracle in Cloud AWS
- **Riacquistare (drop and shop)**: passa a un prodotto diverso, in genere effettuando la transizione da una licenza tradizionale a un modello SaaS. Esempio: migra il tuo sistema di gestione delle relazioni con i clienti (CRM) su Salesforce.com
- **Eseguire il rehosting (lift and shift)**: trasferisci un'applicazione sul cloud senza apportare modifiche per sfruttare le funzionalità del cloud. Esempio: migra il tuo database Oracle locale su Oracle su un'istanza EC2 in Cloud AWS
- **Trasferire (eseguire il rehosting a livello hypervisor)**: trasferisci l'infrastruttura sul cloud senza acquistare nuovo hardware, riscrivere le applicazioni o modificare le operazioni esistenti. Esegui la migrazione dei server da una piattaforma locale a un servizio cloud per la stessa piattaforma. Esempio: migra un'applicazione su Microsoft Hyper-V. AWS
- **Riesaminare (mantenere)**: mantieni le applicazioni nell'ambiente di origine. Queste potrebbero includere applicazioni che richiedono una rifattorizzazione significativa che desideri rimandare a un momento successivo e applicazioni legacy che desideri mantenere, perché non vi è alcuna giustificazione aziendale per effettuarne la migrazione.
- **Ritirare**: disattiva o rimuovi le applicazioni che non sono più necessarie nell'ambiente di origine.

# A

## A2A () Agent-to-Agent

Un protocollo statico per la collaborazione tra agenti che supporta la delega delle attività e il trasferimento dello stato.

## ABAC

[Vedi controllo degli accessi basato sugli attributi.](#)

## servizi astratti

Vedi [servizi gestiti](#).

## ACIDO

Vedi [atomicità, consistenza, isolamento, durata](#).

## migrazione attiva-attiva

Un metodo di migrazione del database in cui i database di origine e di destinazione vengono mantenuti sincronizzati (utilizzando uno strumento di replica bidirezionale o operazioni di doppia scrittura) ed entrambi i database gestiscono le transazioni provenienti dalle applicazioni di connessione durante la migrazione. Questo metodo supporta la migrazione in piccoli batch controllati anziché richiedere una conversione una tantum. È più flessibile ma richiede più lavoro rispetto alla migrazione [attiva-passiva](#).

## migrazione attiva-passiva

Un metodo di migrazione del database in cui i database di origine e di destinazione vengono mantenuti sincronizzati, ma solo il database di origine gestisce le transazioni provenienti dalle applicazioni di connessione mentre i dati vengono replicati nel database di destinazione. Il database di destinazione non accetta alcuna transazione durante la migrazione.

## Agente

Un sistema di intelligenza artificiale in grado di ragionare, pianificare e intraprendere azioni in modo autonomo utilizzando strumenti per raggiungere gli obiettivi.

## Agente Ops

Pratiche operative per la creazione, il test, l'implementazione e l'esecuzione di agenti di intelligenza artificiale in produzione su larga scala.

## funzione aggregata

Una funzione SQL che opera su un gruppo di righe e calcola un singolo valore restituito per il gruppo. Esempi di funzioni aggregate includono SUM e MAX.

## Intelligenza artificiale

Vedi [intelligenza artificiale](#).

## AIOps

Guarda le [operazioni di intelligenza artificiale](#).

## anonimizzazione

Il processo di eliminazione permanente delle informazioni personali in un set di dati.

L'anonimizzazione può aiutare a proteggere la privacy personale. I dati anonimi non sono più considerati dati personali.

## anti-modello

Una soluzione utilizzata frequentemente per un problema ricorrente in cui la soluzione è controproducente, inefficace o meno efficace di un'alternativa.

## controllo delle applicazioni

Un approccio alla sicurezza che consente l'uso solo di applicazioni approvate per proteggere un sistema dal malware.

## portfolio di applicazioni

Una raccolta di informazioni dettagliate su ogni applicazione utilizzata da un'organizzazione, compresi i costi di creazione e manutenzione dell'applicazione e il relativo valore aziendale. Queste informazioni sono fondamentali per [il processo di scoperta e analisi del portfolio](#) e aiutano a identificare e ad assegnare la priorità alle applicazioni da migrare, modernizzare e ottimizzare.

## intelligenza artificiale (IA)

Il campo dell'informatica dedicato all'uso delle tecnologie informatiche per svolgere funzioni cognitive tipicamente associate agli esseri umani, come l'apprendimento, la risoluzione di problemi e il riconoscimento di schemi. Per ulteriori informazioni, consulta la sezione [Che cos'è l'intelligenza artificiale?](#)

## operazioni di intelligenza artificiale (AIOps)

Il processo di utilizzo delle tecniche di machine learning per risolvere problemi operativi, ridurre gli incidenti operativi e l'intervento umano e aumentare la qualità del servizio. Per ulteriori

informazioni su come viene utilizzato AIOps nella strategia di migrazione AWS , consulta la [guida all'integrazione delle operazioni](#).

### crittografia asimmetrica

Un algoritmo di crittografia che utilizza una coppia di chiavi, una chiave pubblica per la crittografia e una chiave privata per la decrittografia. Puoi condividere la chiave pubblica perché non viene utilizzata per la decrittografia, ma l'accesso alla chiave privata deve essere altamente limitato.

### atomicità, consistenza, isolamento, durabilità (ACID)

Un insieme di proprietà del software che garantiscono la validità dei dati e l'affidabilità operativa di un database, anche in caso di errori, interruzioni di corrente o altri problemi.

### Controllo degli accessi basato su attributi (ABAC)

La pratica di creare autorizzazioni dettagliate basate su attributi utente, come reparto, ruolo professionale e nome del team. Per ulteriori informazioni, consulta [ABAC for AWS](#) nella documentazione AWS Identity and Access Management (IAM).

### fonte di dati autorevole

Una posizione in cui è archiviata la versione principale dei dati, considerata la fonte di informazioni più affidabile. È possibile copiare i dati dalla fonte di dati autorevole in altre posizioni allo scopo di elaborarli o modificarli, ad esempio anonimizzandoli, oscurandoli o pseudonimizzandoli.

### Zona di disponibilità

Una posizione distinta all'interno di un edificio Regione AWS che è isolata dai guasti in altre zone di disponibilità e offre una connettività di rete economica e a bassa latenza verso altre zone di disponibilità nella stessa regione.

### AWS Cloud Adoption Framework (CAF)AWS

Un framework di linee guida e best practice AWS per aiutare le organizzazioni a sviluppare un piano efficiente ed efficace per passare con successo al cloud. AWS CAF organizza le linee guida in sei aree di interesse chiamate prospettive: business, persone, governance, piattaforma, sicurezza e operazioni. Le prospettive relative ad azienda, persone e governance si concentrano sulle competenze e sui processi aziendali; le prospettive relative alla piattaforma, alla sicurezza e alle operazioni si concentrano sulle competenze e sui processi tecnici. Ad esempio, la prospettiva relativa alle persone si rivolge alle parti interessate che gestiscono le risorse umane (HR), le funzioni del personale e la gestione del personale. In questa prospettiva, AWS CAF fornisce linee guida per lo sviluppo delle persone, la formazione e le comunicazioni per aiutare a preparare

l'organizzazione all'adozione del cloud di successo. Per ulteriori informazioni, consulta il [sito web di AWS CAF](#) e il [white paper AWS CAF](#).

## AWS Workload Qualification Framework (WQF)AWS

Uno strumento che valuta i carichi di lavoro di migrazione dei database, consiglia strategie di migrazione e fornisce stime del lavoro. AWS WQF è incluso in (). AWS Schema Conversion Tool AWS SCT Analizza gli schemi di database e gli oggetti di codice, il codice dell'applicazione, le dipendenze e le caratteristiche delle prestazioni e fornisce report di valutazione.

## B

### bot difettoso

Un [bot](#) che ha lo scopo di disturbare o causare danni a individui o organizzazioni.

### BCP

Vedi la [pianificazione della continuità operativa](#).

### grafico comportamentale

Una vista unificata, interattiva dei comportamenti delle risorse e delle interazioni nel tempo. Puoi utilizzare un grafico comportamentale con Amazon Detective per esaminare tentativi di accesso non riusciti, chiamate API sospette e azioni simili. Per ulteriori informazioni, consulta [Dati in un grafico comportamentale](#) nella documentazione di Detective.

### sistema big-endian

Un sistema che memorizza per primo il byte più importante. Vedi anche [endianness](#).

### Classificazione binaria

Un processo che prevede un risultato binario (una delle due classi possibili). Ad esempio, il modello di machine learning potrebbe dover prevedere problemi come "Questa e-mail è spam o non è spam?" o "Questo prodotto è un libro o un'auto?"

### filtro Bloom

Una struttura di dati probabilistica ed efficiente in termini di memoria che viene utilizzata per verificare se un elemento fa parte di un set.

## blue/green dispiegamento

Una strategia di implementazione in cui si creano due ambienti separati ma identici. La versione corrente dell'applicazione viene eseguita in un ambiente (blu) e la nuova versione dell'applicazione nell'altro ambiente (verde). Questa strategia consente di ripristinare rapidamente il sistema con un impatto minimo.

## bot

Un'applicazione software che esegue attività automatizzate su Internet e simula l'attività o l'interazione umana. Alcuni bot sono utili o utili, come i web crawler che indicizzano le informazioni su Internet. Alcuni altri bot, noti come bot dannosi, hanno lo scopo di disturbare o causare danni a individui o organizzazioni.

## botnet

Reti di [bot](#) infettate da [malware](#) e controllate da un'unica parte, nota come bot herder o bot operator. Le botnet sono il meccanismo più noto per scalare i bot e il loro impatto.

## ramo

Un'area contenuta di un repository di codice. Il primo ramo creato in un repository è il ramo principale. È possibile creare un nuovo ramo a partire da un ramo esistente e quindi sviluppare funzionalità o correggere bug al suo interno. Un ramo creato per sviluppare una funzionalità viene comunemente detto ramo di funzionalità. Quando la funzionalità è pronta per il rilascio, il ramo di funzionalità viene ricongiunto al ramo principale. Per ulteriori informazioni, consulta [Informazioni sulle filiali](#) (documentazione). GitHub

## accesso break-glass

In circostanze eccezionali e tramite una procedura approvata, un mezzo rapido per consentire a un utente di accedere a un sito a Account AWS cui in genere non dispone delle autorizzazioni necessarie. Per ulteriori informazioni, consulta l'indicatore [Implementare le procedure break-glass](#) nella guida. AWS Well-Architected

## strategia brownfield

L'infrastruttura esistente nell'ambiente. Quando si adotta una strategia brownfield per un'architettura di sistema, si progetta l'architettura in base ai vincoli dei sistemi e dell'infrastruttura attuali. Per l'espansione dell'infrastruttura esistente, è possibile combinare strategie brownfield e [greenfield](#).

## cache del buffer

L'area di memoria in cui sono archiviati i dati a cui si accede con maggiore frequenza.

## capacità di business

Azioni intraprese da un'azienda per generare valore (ad esempio vendite, assistenza clienti o marketing). Le architetture dei microservizi e le decisioni di sviluppo possono essere guidate dalle capacità aziendali. Per ulteriori informazioni, consulta la sezione [Organizzazione in base alle funzionalità aziendali](#) del whitepaper [Esecuzione di microservizi containerizzati su AWS](#).

## pianificazione della continuità operativa (BCP)

Un piano che affronta il potenziale impatto di un evento che comporta l'interruzione dell'attività, come una migrazione su larga scala, sulle operazioni e consente a un'azienda di riprendere rapidamente le operazioni.

# C

## CAF

Vedi [AWS Cloud Adoption Framework](#).

## implementazione canaria

Il rilascio lento e incrementale di una versione agli utenti finali. Quando sei sicuro, distribuisce la nuova versione e sostituisci la versione corrente nella sua interezza.

## CoE

Vedi [Cloud Center of Excellence](#).

## CDC

Vedi [Change Data Capture](#).

## Change Data Capture (CDC)

Il processo di tracciamento delle modifiche a un'origine dati, ad esempio una tabella di database, e di registrazione dei metadati relativi alla modifica. È possibile utilizzare CDC per vari scopi, ad esempio il controllo o la replica delle modifiche in un sistema di destinazione per mantenere la sincronizzazione.

## ingegneria del caos

Introduzione intenzionale di guasti o eventi dirompenti per testare la resilienza di un sistema. Puoi usare [AWS Fault Injection Service \(AWS FIS\)](#) per eseguire esperimenti che stressano i tuoi AWS carichi di lavoro e valutarne la risposta.

## CI/CD

Vedi [integrazione continua e distribuzione continua](#).

## classificazione

Un processo di categorizzazione che aiuta a generare previsioni. I modelli di ML per problemi di classificazione prevedono un valore discreto. I valori discreti sono sempre distinti l'uno dall'altro. Ad esempio, un modello potrebbe dover valutare se in un'immagine è presente o meno un'auto.

## Sviluppatore cittadino

Un utente aziendale che crea applicazioni di intelligenza artificiale utilizzando piattaforme senza code/low codice senza competenze tecniche specializzate.

## crittografia lato client

Crittografia dei dati localmente, prima che il bersaglio li Servizio AWS riceva.

## centro di eccellenza del cloud (CCoE)

Un team multidisciplinare che guida le iniziative di adozione del cloud in tutta l'organizzazione, tra cui lo sviluppo di best practice per il cloud, la mobilitazione delle risorse, la definizione delle tempistiche di migrazione e la guida dell'organizzazione attraverso trasformazioni su larga scala. Per ulteriori informazioni, consulta i [post di CCoE](#) sull' Cloud AWS Enterprise Strategy Blog.

## cloud computing

La tecnologia cloud generalmente utilizzata per l'archiviazione remota di dati e la gestione dei dispositivi IoT. Il cloud computing è generalmente collegato alla tecnologia di [edge computing](#).

## modello operativo cloud

In un'organizzazione IT, il modello operativo utilizzato per creare, maturare e ottimizzare uno o più ambienti cloud. Per ulteriori informazioni, consulta [Building your Cloud Operating Model](#).

## fasi di adozione del cloud

Le quattro fasi che le organizzazioni in genere attraversano quando migrano verso Cloud AWS:

- Progetto: esecuzione di alcuni progetti relativi al cloud per scopi di dimostrazione e apprendimento
- Fondamento: effettuare investimenti fondamentali per dimensionare l'adozione del cloud (ad esempio, creazione di una zona di destinazione, definizione di un CCoE, definizione di un modello operativo)
- Migrazione: migrazione di singole applicazioni

- Re-invention — Ottimizzazione di prodotti e servizi e innovazione nel cloud

Queste fasi sono state definite da Stephen Orban nel post del blog [The Journey Toward Cloud-First & the Stages of Adoption](#) sul blog Enterprise Strategy. Cloud AWS Per informazioni sulla loro relazione con la strategia di AWS migrazione, consulta la guida alla [preparazione alla migrazione](#).

## CMDB

Vedi [database di gestione della configurazione](#).

## repository di codice

Una posizione in cui il codice di origine e altri asset, come documentazione, esempi e script, vengono archiviati e aggiornati attraverso processi di controllo delle versioni. Gli archivi cloud più comuni includono GitHub o Bitbucket Cloud. Ogni versione del codice è denominata ramo. In una struttura a microservizi, ogni repository è dedicato a una singola funzionalità. Una singola CI/CD pipeline può utilizzare più repository.

## cache fredda

Una cache del buffer vuota, non ben popolata o contenente dati obsoleti o irrilevanti. Ciò influisce sulle prestazioni perché l'istanza di database deve leggere dalla memoria o dal disco principale, il che richiede più tempo rispetto alla lettura dalla cache del buffer.

## dati freddi

Dati a cui si accede raramente e che in genere sono storici. Quando si eseguono interrogazioni di questo tipo di dati, le interrogazioni lente sono in genere accettabili. Lo spostamento di questi dati su livelli o classi di storage meno costosi e con prestazioni inferiori può ridurre i costi.

## visione artificiale (CV)

Un campo dell'[intelligenza artificiale](#) che utilizza l'apprendimento automatico per analizzare ed estrarre informazioni da formati visivi come immagini e video digitali. Ad esempio, Amazon SageMaker AI fornisce algoritmi di elaborazione delle immagini per CV.

## deriva della configurazione

Per un carico di lavoro, una modifica della configurazione rispetto allo stato previsto. Potrebbe causare la non conformità del carico di lavoro e in genere è graduale e involontaria.

## database di gestione della configurazione (CMDB)

Un repository che archivia e gestisce le informazioni su un database e il relativo ambiente IT, inclusi i componenti hardware e software e le relative configurazioni. In genere si utilizzano i dati di un CMDB nella fase di individuazione e analisi del portafoglio della migrazione.

## Pacchetto di conformità

Una raccolta di AWS Config regole e azioni correttive che puoi assemblare per personalizzare i controlli di conformità e sicurezza. È possibile distribuire un pacchetto di conformità come singola entità in una regione Account AWS and o all'interno di un'organizzazione utilizzando un modello YAML. Per ulteriori informazioni, consulta i [Conformance](#) Pack nella documentazione. AWS Config

## integrazione e distribuzione continue ( ) CI/CD

Il processo di automazione delle fasi di origine, compilazione, test, gestione temporanea e produzione del processo di rilascio del software. CI/CD viene comunemente descritto come una pipeline. CI/CD può aiutarvi ad automatizzare i processi, migliorare la produttività, migliorare la qualità del codice e velocizzare le consegne. Per ulteriori informazioni, consulta [Vantaggi della distribuzione continua](#). CD può anche significare continuous deployment (implementazione continua). Per ulteriori informazioni, consulta [Distribuzione continua e implementazione continua a confronto](#).

## CV

Vedi [visione artificiale](#).

## D

### dati a riposo

Dati stazionari nella rete, ad esempio i dati archiviati.

### classificazione dei dati

Un processo per identificare e classificare i dati nella rete in base alla loro criticità e sensibilità. È un componente fondamentale di qualsiasi strategia di gestione dei rischi di sicurezza informatica perché consente di determinare i controlli di protezione e conservazione appropriati per i dati. La classificazione dei dati è un componente del pilastro della sicurezza nel AWS Well-Architected Framework. Per ulteriori informazioni, consulta [Classificazione dei dati](#).

### deriva dei dati

Una variazione significativa tra i dati di produzione e i dati utilizzati per addestrare un modello di machine learning o una modifica significativa dei dati di input nel tempo. La deriva dei dati può ridurre la qualità, l'accuratezza e l'equità complessive nelle previsioni dei modelli ML.

## dati in transito

Dati che si spostano attivamente attraverso la rete, ad esempio tra le risorse di rete.

## rete di dati

Un framework architettonico che fornisce la proprietà distribuita e decentralizzata dei dati con gestione e governance centralizzate.

## riduzione al minimo dei dati

Il principio della raccolta e del trattamento dei soli dati strettamente necessari. Praticare la riduzione al minimo dei dati in the Cloud AWS può ridurre i rischi per la privacy, i costi e l'impronta di carbonio delle analisi.

## perimetro dei dati

Una serie di barriere preventive nell' AWS ambiente che aiutano a garantire che solo le identità attendibili accedano alle risorse attendibili delle reti previste. Per ulteriori informazioni, consulta [Building a data perimeter](#) on AWS.

## pre-elaborazione dei dati

Trasformare i dati grezzi in un formato che possa essere facilmente analizzato dal modello di ML. La pre-elaborazione dei dati può comportare la rimozione di determinate colonne o righe e l'eliminazione di valori mancanti, incoerenti o duplicati.

## provenienza dei dati

Il processo di tracciamento dell'origine e della cronologia dei dati durante il loro ciclo di vita, ad esempio il modo in cui i dati sono stati generati, trasmessi e archiviati.

## soggetto dei dati

Un individuo i cui dati vengono raccolti ed elaborati.

## data warehouse

Un sistema di gestione dei dati che supporta la business intelligence, come l'analisi. I data warehouse contengono in genere grandi quantità di dati storici e vengono generalmente utilizzati per interrogazioni e analisi.

## linguaggio di definizione del database (DDL)

Istruzioni o comandi per creare o modificare la struttura di tabelle e oggetti in un database.

## linguaggio di manipolazione del database (DML)

Istruzioni o comandi per modificare (inserire, aggiornare ed eliminare) informazioni in un database.

## DDL

Vedi linguaggio di [definizione del database](#).

## deep ensemble

Combinare più modelli di deep learning per la previsione. È possibile utilizzare i deep ensemble per ottenere una previsione più accurata o per stimare l'incertezza nelle previsioni.

## deep learning

Un sottocampo del ML che utilizza più livelli di reti neurali artificiali per identificare la mappatura tra i dati di input e le variabili target di interesse.

## difesa in profondità

Un approccio alla sicurezza delle informazioni in cui una serie di meccanismi e controlli di sicurezza sono accuratamente stratificati su una rete di computer per proteggere la riservatezza, l'integrità e la disponibilità della rete e dei dati al suo interno. Quando si adotta questa strategia AWS, si aggiungono più controlli a diversi livelli della AWS Organizations struttura per proteggere le risorse. Ad esempio, un approccio di difesa approfondita potrebbe combinare autenticazione a più fattori, segmentazione della rete e crittografia.

## amministratore delegato

In AWS Organizations, un servizio compatibile può registrare un account AWS membro per amministrare gli account dell'organizzazione e gestire le autorizzazioni per quel servizio. Questo account è denominato amministratore delegato per quel servizio specifico. Per ulteriori informazioni e un elenco di servizi compatibili, consulta [Servizi che funzionano con AWS Organizations](#) nella documentazione di AWS Organizations .

## implementazione

Il processo di creazione di un'applicazione, di nuove funzionalità o di correzioni di codice disponibili nell'ambiente di destinazione. L'implementazione prevede l'applicazione di modifiche in una base di codice, seguita dalla creazione e dall'esecuzione di tale base di codice negli ambienti applicativi.

## Ambiente di sviluppo

[Vedi ambiente](#).

## controllo di rilevamento

Un controllo di sicurezza progettato per rilevare, registrare e avvisare dopo che si è verificato un evento. Questi controlli rappresentano una seconda linea di difesa e avvisano l'utente in caso di eventi di sicurezza che aggirano i controlli preventivi in vigore. Per ulteriori informazioni, consulta [Controlli di rilevamento](#) in Implementazione dei controlli di sicurezza in AWS.

## mappatura del flusso di valore dello sviluppo (DVSM)

Un processo utilizzato per identificare e dare priorità ai vincoli che influiscono negativamente sulla velocità e sulla qualità nel ciclo di vita dello sviluppo del software. DVSM estende il processo di mappatura del flusso di valore originariamente progettato per pratiche di produzione snella. Si concentra sulle fasi e sui team necessari per creare e trasferire valore attraverso il processo di sviluppo del software.

## gemello digitale

Una rappresentazione virtuale di un sistema reale, ad esempio un edificio, una fabbrica, un'attrezzatura industriale o una linea di produzione. I gemelli digitali supportano la manutenzione predittiva, il monitoraggio remoto e l'ottimizzazione della produzione.

## tabella delle dimensioni

In uno [schema a stella](#), una tabella più piccola che contiene gli attributi dei dati quantitativi in una tabella dei fatti. Gli attributi della tabella delle dimensioni sono in genere campi di testo o numeri discreti che si comportano come testo. Questi attributi vengono comunemente utilizzati per il vincolo delle query, il filtraggio e l'etichettatura dei set di risultati.

## disastro

Un evento che impedisce a un carico di lavoro o a un sistema di raggiungere gli obiettivi aziendali nella sua sede principale di implementazione. Questi eventi possono essere disastri naturali, guasti tecnici o il risultato di azioni umane, come errori di configurazione involontari o attacchi di malware.

## disaster recovery (DR)

La strategia e il processo utilizzati per ridurre al minimo i tempi di inattività e la perdita di dati causati da un [disastro](#). Per ulteriori informazioni, consulta [Disaster Recovery of Workload su AWS: Recovery in the Cloud in the](#) AWS Well-Architected Framework.

## DML

Vedi linguaggio di [manipolazione del database](#).

## progettazione basata sul dominio

Un approccio allo sviluppo di un sistema software complesso collegandone i componenti a domini in evoluzione, o obiettivi aziendali principali, perseguiti da ciascun componente. Questo concetto è stato introdotto da Eric Evans nel suo libro *Domain-Driven Design: Tackling Complexity in the Heart of Software* (Boston: Addison-Wesley Professional, 2003). Per informazioni su come utilizzare la progettazione basata sul dominio con lo strangler fig pattern, consulta [Modernizzare i servizi Web Microsoft ASP.NET \(ASMX\) legacy in modo incrementale utilizzando contenitori e Amazon API Gateway](#).

## DOTT.

Vedi [disaster recovery](#).

## rilevamento della deriva

Tracciamento delle deviazioni da una configurazione di base. Ad esempio, puoi utilizzarlo AWS CloudFormation per [rilevare la deriva nelle risorse di sistema](#) oppure puoi usarlo AWS Control Tower per [rilevare cambiamenti nella tua landing zone](#) che potrebbero influire sulla conformità ai requisiti di governance.

## DVSM

Vedi la [mappatura del flusso di valore dello sviluppo](#).

## E

### EDA

Vedi [analisi esplorativa dei dati](#).

### MODIFICA

Vedi [scambio elettronico di dati](#).

### edge computing

La tecnologia che aumenta la potenza di calcolo per i dispositivi intelligenti all'edge di una rete IoT. Rispetto al [cloud computing](#), [l'edge computing](#) può ridurre la latenza di comunicazione e migliorare i tempi di risposta.

### scambio elettronico di dati (EDI)

Lo scambio automatizzato di documenti aziendali tra organizzazioni. Per ulteriori informazioni, vedere [Cos'è lo scambio elettronico di dati](#).

## crittografia

Un processo di elaborazione che trasforma i dati in chiaro, leggibili dall'uomo, in testo cifrato.

### chiave crittografica

Una stringa crittografica di bit randomizzati generata da un algoritmo di crittografia. Le chiavi possono variare di lunghezza e ogni chiave è progettata per essere imprevedibile e univoca.

### endianità

L'ordine in cui i byte vengono archiviati nella memoria del computer. Big-endian i sistemi memorizzano per primi il byte più importante. Little-endian i sistemi memorizzano per primi il byte meno importante.

### endpoint

Vedi [service endpoint](#).

### servizio endpoint

Un servizio che puoi ospitare in un cloud privato virtuale (VPC) da condividere con altri utenti. Puoi creare un servizio endpoint con AWS PrivateLink e concedere autorizzazioni ad altri Account AWS o a AWS Identity and Access Management (IAM) principali. Questi account o principali possono connettersi al servizio endpoint in privato creando endpoint VPC di interfaccia. Per ulteriori informazioni, consulta [Creazione di un servizio endpoint](#) nella documentazione di Amazon Virtual Private Cloud (Amazon VPC).

### pianificazione delle risorse aziendali (ERP)

Un sistema che automatizza e gestisce i processi aziendali chiave (come contabilità, [MES](#) e gestione dei progetti) per un'azienda.

### crittografia envelope

Il processo di crittografia di una chiave di crittografia con un'altra chiave di crittografia. Per ulteriori informazioni, vedete [Envelope encryption](#) nella documentazione AWS Key Management Service (AWS KMS).

### ambiente

Un'istanza di un'applicazione in esecuzione. Di seguito sono riportati i tipi di ambiente più comuni nel cloud computing:

- ambiente di sviluppo: un'istanza di un'applicazione in esecuzione disponibile solo per il team principale responsabile della manutenzione dell'applicazione. Gli ambienti di sviluppo vengono

utilizzati per testare le modifiche prima di promuoverle negli ambienti superiori. Questo tipo di ambiente viene talvolta definito ambiente di test.

- ambienti inferiori: tutti gli ambienti di sviluppo di un'applicazione, ad esempio quelli utilizzati per le build e i test iniziali.
- ambiente di produzione: un'istanza di un'applicazione in esecuzione a cui gli utenti finali possono accedere. In una CI/CD pipeline, l'ambiente di produzione è l'ultimo ambiente di distribuzione.
- ambienti superiori: tutti gli ambienti a cui possono accedere utenti diversi dal team di sviluppo principale. Si può trattare di un ambiente di produzione, ambienti di preproduzione e ambienti per i test di accettazione da parte degli utenti.

## epica

Nelle metodologie agili, categorie funzionali che aiutano a organizzare e dare priorità al lavoro. Le epiche forniscono una descrizione di alto livello dei requisiti e delle attività di implementazione. Ad esempio, le epiche della sicurezza AWS CAF includono la gestione delle identità e degli accessi, i controlli investigativi, la sicurezza dell'infrastruttura, la protezione dei dati e la risposta agli incidenti. Per ulteriori informazioni sulle epiche, consulta la strategia di migrazione AWS , consulta la [guida all'implementazione del programma](#).

## ERP

Vedi [pianificazione delle risorse aziendali](#).

## analisi esplorativa dei dati (EDA)

Il processo di analisi di un set di dati per comprenderne le caratteristiche principali. Si raccolgono o si aggregano dati e quindi si eseguono indagini iniziali per trovare modelli, rilevare anomalie e verificare ipotesi. L'EDA viene eseguita calcolando statistiche di riepilogo e creando visualizzazioni di dati.

## F

### tabella dei fatti

Il tavolo centrale con [schema a stella](#). Memorizza dati quantitativi sulle operazioni aziendali. In genere, una tabella dei fatti contiene due tipi di colonne: quelle che contengono misure e quelle che contengono una chiave esterna per una tabella di dimensioni.

## fallire velocemente

Una filosofia che utilizza test frequenti e incrementali per ridurre il ciclo di vita dello sviluppo. È una parte fondamentale di un approccio agile.

## limite di isolamento dei guasti

Nel Cloud AWS, un limite come una zona di disponibilità Regione AWS, un piano di controllo o un piano dati che limita l'effetto di un errore e aiuta a migliorare la resilienza dei carichi di lavoro. Per ulteriori informazioni, consulta [AWS Fault Isolation Boundaries](#).

## ramo di funzionalità

Vedi [filiale](#).

## caratteristiche

I dati di input che usi per fare una previsione. Ad esempio, in un contesto di produzione, le caratteristiche potrebbero essere immagini acquisite periodicamente dalla linea di produzione.

## importanza delle caratteristiche

Quanto è importante una caratteristica per le previsioni di un modello. Di solito viene espresso come punteggio numerico che può essere calcolato con varie tecniche, come Shapley Additive Explanations (SHAP) e gradienti integrati. Per ulteriori informazioni, consulta [Interpretabilità del modello di machine learning con AWS](#).

## trasformazione delle funzionalità

Per ottimizzare i dati per il processo di machine learning, incluso l'arricchimento dei dati con fonti aggiuntive, il dimensionamento dei valori o l'estrazione di più set di informazioni da un singolo campo di dati. Ciò consente al modello di ML di trarre vantaggio dai dati. Ad esempio, se suddividi la data "2021-05-27 00:15:37" in "2021", "maggio", "giovedì" e "15", puoi aiutare l'algoritmo di apprendimento ad apprendere modelli sfumati associati a diversi componenti dei dati.

## prompt con pochi scatti

Fornire a un [LLM](#) un numero limitato di esempi che dimostrino l'attività e il risultato desiderato prima di chiedergli di eseguire un'attività simile. Questa tecnica è un'applicazione dell'apprendimento contestuale, in cui i modelli imparano da esempi (immagini) incorporati nei prompt. Few-shot i suggerimenti possono essere efficaci per attività che richiedono una formattazione, un ragionamento o una conoscenza del dominio specifici. [Vedi anche zero-shot prompting](#).

## FGAC

Vedi il controllo [granulare degli accessi](#).

controllo granulare degli accessi (FGAC)

L'uso di più condizioni per consentire o rifiutare una richiesta di accesso.

migrazione flash-cut

Un metodo di migrazione del database che utilizza la replica continua dei dati tramite [l'acquisizione dei dati delle modifiche](#) per migrare i dati nel più breve tempo possibile, anziché utilizzare un approccio graduale. L'obiettivo è ridurre al minimo i tempi di inattività.

## FM

[Vedi il modello di base.](#)

modello di fondazione (FM)

Una grande rete neurale di deep learning che si è addestrata su enormi set di dati generalizzati e non etichettati. Le FM sono in grado di eseguire un'ampia varietà di attività generali, come comprendere il linguaggio, generare testo e immagini e conversare in linguaggio naturale. Per ulteriori informazioni, consulta [Cosa sono i modelli Foundation](#).

Gateway FM

[Un intermediario centralizzato che controlla e normalizza l'accesso ai modelli di base.](#) Conosciuto anche come gateway LLM.

## G

IA generativa

Un sottoinsieme di modelli di [intelligenza artificiale](#) che sono stati addestrati su grandi quantità di dati e che possono utilizzare un semplice messaggio di testo per creare nuovi contenuti e artefatti, come immagini, video, testo e audio. Per ulteriori informazioni, consulta [Cos'è l'IA generativa](#).

blocco geografico

Vedi [restrizioni geografiche](#).

limitazioni geografiche (blocco geografico)

In Amazon CloudFront, un'opzione per impedire agli utenti di determinati paesi di accedere alle distribuzioni di contenuti. Puoi utilizzare un elenco consentito o un elenco di blocco per specificare

i paesi approvati e vietati. Per ulteriori informazioni, consulta [Limitare la distribuzione geografica dei contenuti](#) nella CloudFront documentazione.

## Flusso di lavoro di GitFlow

Un approccio in cui gli ambienti inferiori e superiori utilizzano rami diversi in un repository di codice di origine. Il flusso di lavoro Gitflow è considerato obsoleto e il flusso di lavoro [basato su trunk è l'approccio moderno e preferito](#).

## immagine dorata

Un'istantanea di un sistema o di un software che viene utilizzata come modello per distribuire nuove istanze di quel sistema o software. Ad esempio, nella produzione, un'immagine dorata può essere utilizzata per fornire software su più dispositivi e contribuire a migliorare la velocità, la scalabilità e la produttività nelle operazioni di produzione dei dispositivi.

## strategia greenfield

L'assenza di infrastrutture esistenti in un nuovo ambiente. Quando si adotta una strategia greenfield per un'architettura di sistema, è possibile selezionare tutte le nuove tecnologie senza il vincolo della compatibilità con l'infrastruttura esistente, nota anche come [brownfield](#). Per l'espansione dell'infrastruttura esistente, è possibile combinare strategie brownfield e greenfield.

## guardrail

Una regola di livello elevato che consente di governare risorse, policy e conformità tra le unità organizzative (OU). I guardrail preventivi applicano le policy per garantire l'allineamento agli standard di conformità. Vengono implementati utilizzando le policy di controllo dei servizi e i limiti delle autorizzazioni IAM. I guardrail di rilevamento rilevano le violazioni delle policy e i problemi di conformità e generano avvisi per porvi rimedio. Sono implementati utilizzando Amazon AWS Config AWS Security Hub CSPM GuardDuty AWS Trusted Advisor, Amazon Inspector e controlli personalizzati AWS Lambda .

## guardrail (AI)

Meccanismi di sicurezza che filtrano, convalidano e limitano gli input e gli output degli [agenti](#) per contribuire a garantire un comportamento dell'IA responsabile e sicuro.

# H

## AH

Vedi [disponibilità elevata](#).

## migrazione di database eterogenea

Migrazione del database di origine in un database di destinazione che utilizza un motore di database diverso (ad esempio, da Oracle ad Amazon Aurora). La migrazione eterogenea fa in genere parte di uno sforzo di riprogettazione e la conversione dello schema può essere un'attività complessa. [AWS offre AWS SCT](#) che aiuta con le conversioni dello schema.

## alta disponibilità (HA)

La capacità di un carico di lavoro di funzionare in modo continuo, senza intervento, in caso di sfide o disastri. I sistemi HA sono progettati per il failover automatico, fornire costantemente prestazioni di alta qualità e gestire carichi e guasti diversi con un impatto minimo sulle prestazioni.

## modernizzazione storica

Un approccio utilizzato per modernizzare e aggiornare i sistemi di tecnologia operativa (OT) per soddisfare meglio le esigenze dell'industria manifatturiera. Uno storico è un tipo di database utilizzato per raccogliere e archiviare dati da varie fonti in una fabbrica.

## dati di esclusione

[Una parte di dati storici etichettati che viene trattenuta da un set di dati utilizzata per addestrare un modello di apprendimento automatico.](#) È possibile utilizzare i dati di holdout per valutare le prestazioni del modello confrontando le previsioni del modello con i dati di holdout.

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

Un modello di flusso di lavoro in cui l'esecuzione degli [agenti](#) viene sospesa per la revisione e l'approvazione umana nei punti decisionali critici.

## migrazione di database omogenea

Migrazione del database di origine in un database di destinazione che condivide lo stesso motore di database (ad esempio, da Microsoft SQL Server ad Amazon RDS per SQL Server). La migrazione omogenea fa in genere parte di un'operazione di rehosting o ridefinizione della piattaforma. Per migrare lo schema è possibile utilizzare le utilità native del database.

## dati caldi

Dati a cui si accede frequentemente, ad esempio dati in tempo reale o dati di traduzione recenti. Questi dati richiedono in genere un livello o una classe di storage ad alte prestazioni per fornire risposte rapide alle query.

## hotfix

Una soluzione urgente per un problema critico in un ambiente di produzione. A causa della sua urgenza, un hotfix viene in genere creato al di fuori del tipico DevOps flusso di lavoro di rilascio.

## periodo di hypercare

Subito dopo la conversione, il periodo di tempo in cui un team di migrazione gestisce e monitora le applicazioni migrate nel cloud per risolvere eventuali problemi. In genere, questo periodo dura da 1 a 4 giorni. Al termine del periodo di hypercare, il team addetto alla migrazione in genere trasferisce la responsabilità delle applicazioni al team addetto alle operazioni cloud.

## I

### laC

Vedi [l'infrastruttura come codice](#).

### Policy basata su identità

Una policy associata a uno o più principi IAM che definisce le relative autorizzazioni all'interno dell'Cloud AWS ambiente.

### applicazione inattiva

Un'applicazione che prevede un uso di CPU e memoria medio compreso tra il 5% e il 20% in un periodo di 90 giorni. In un progetto di migrazione, è normale ritirare queste applicazioni o mantenerle on-premise.

## IIoT

Vedi [Industrial Internet of Things](#).

### infrastruttura immutabile

Un modello che implementa una nuova infrastruttura per i carichi di lavoro di produzione anziché aggiornare, applicare patch o modificare l'infrastruttura esistente. [Le infrastrutture immutabili sono intrinsecamente più coerenti, affidabili e prevedibili delle infrastrutture mutabili](#). Per ulteriori informazioni, consulta la best practice [Deploy using immutable](#) infrastructure nel Framework. AWS Well-Architected

### VPC in ingresso (ingress)

In un'architettura AWS multi-account, un VPC che accetta, ispeziona e indirizza le connessioni di rete dall'esterno di un'applicazione. Nel documento [Architettura di riferimento per la sicurezza di](#)

[AWS](#) si consiglia di configurare l'account di rete con VPC in entrata, in uscita e di ispezione per proteggere l'interfaccia bidirezionale tra l'applicazione e Internet in generale.

## migrazione incrementale

Una strategia di conversione in cui si esegue la migrazione dell'applicazione in piccole parti anziché eseguire una conversione singola e completa. Ad esempio, inizialmente potresti spostare solo alcuni microservizi o utenti nel nuovo sistema. Dopo aver verificato che tutto funzioni correttamente, puoi spostare in modo incrementale microservizi o utenti aggiuntivi fino alla disattivazione del sistema legacy. Questa strategia riduce i rischi associati alle migrazioni di grandi dimensioni.

## Industria 4.0

Un termine introdotto da [Klaus Schwab](#) nel 2016 per riferirsi alla modernizzazione dei processi di produzione attraverso progressi in termini di connettività, dati in tempo reale, automazione, analisi e. AI/ML

## infrastruttura

Tutte le risorse e gli asset contenuti nell'ambiente di un'applicazione.

## infrastruttura come codice (IaC)

Il processo di provisioning e gestione dell'infrastruttura di un'applicazione tramite un insieme di file di configurazione. Il processo IaC è progettato per aiutarti a centralizzare la gestione dell'infrastruttura, a standardizzare le risorse e a dimensionare rapidamente, in modo che i nuovi ambienti siano ripetibili, affidabili e coerenti.

## Internet delle cose industriale (IIoT)

L'uso di sensori e dispositivi connessi a Internet nei settori industriali, come quello manifatturiero, energetico, automobilistico, sanitario, delle scienze della vita e dell'agricoltura. Per ulteriori informazioni, consulta [Creazione di una strategia di trasformazione digitale dell'Internet delle cose industriale \(IIoT\)](#).

## VPC di ispezione

In un'architettura AWS multi-account, un VPC centralizzato che gestisce le ispezioni del traffico di rete tra VPC (uguali o diversi Regioni AWS), Internet e reti locali. Nel documento [Architettura di riferimento per la sicurezza di AWS](#) si consiglia di configurare l'account di rete con VPC in entrata, in uscita e di ispezione per proteggere l'interfaccia bidirezionale tra l'applicazione e Internet in generale.

## Internet of Things (IoT)

La rete di oggetti fisici connessi con sensori o processori incorporati che comunicano con altri dispositivi e sistemi tramite Internet o una rete di comunicazione locale. Per ulteriori informazioni, consulta [Cos'è l'IoT?](#)

## interpretabilità

Una caratteristica di un modello di machine learning che descrive il grado in cui un essere umano è in grado di comprendere in che modo le previsioni del modello dipendono dai suoi input. [Per ulteriori informazioni, consulta Interpretabilità del modello di machine learning con. AWS](#)

## IoT

Vedi [Internet of Things](#).

## libreria di informazioni IT (ITIL)

Una serie di best practice per offrire servizi IT e allinearli ai requisiti aziendali. ITIL fornisce le basi per ITSM.

## gestione dei servizi IT (ITSM)

Attività associate alla progettazione, implementazione, gestione e supporto dei servizi IT per un'organizzazione. Per informazioni sull'integrazione delle operazioni cloud con gli strumenti ITSM, consulta la [guida all'integrazione delle operazioni](#).

## ITIL

Vedi la [libreria di informazioni IT](#).

## ITSM

Vedi [Gestione dei servizi IT](#).

## L

## controllo degli accessi basato su etichette (LBAC)

Un'implementazione del controllo di accesso obbligatorio (MAC) in cui agli utenti e ai dati stessi viene assegnato esplicitamente un valore di etichetta di sicurezza. L'intersezione tra l'etichetta di sicurezza utente e l'etichetta di sicurezza dei dati determina quali righe e colonne possono essere visualizzate dall'utente.

## zona di destinazione

Una landing zone è un AWS ambiente multi-account ben progettato, scalabile e sicuro. Questo è un punto di partenza dal quale le organizzazioni possono avviare e distribuire rapidamente carichi di lavoro e applicazioni con fiducia nel loro ambiente di sicurezza e infrastruttura. Per ulteriori informazioni sulle zone di destinazione, consulta la sezione [Configurazione di un ambiente AWS multi-account sicuro e scalabile](#).

## modello linguistico di grandi dimensioni (LLM)

Un modello di [intelligenza artificiale](#) di deep learning preaddestrato su una grande quantità di dati. Un LLM può svolgere più attività, come rispondere a domande, riepilogare documenti, tradurre testo in altre lingue e completare frasi. [Per ulteriori informazioni, consulta Cosa sono gli LLM](#).

## migrazione su larga scala

Una migrazione di 300 o più server.

## BIANCO

Vedi controllo degli accessi [basato su etichette](#).

## Privilegio minimo

La best practice di sicurezza per la concessione delle autorizzazioni minime richieste per eseguire un'attività. Per ulteriori informazioni, consulta [Applicazione delle autorizzazioni del privilegio minimo](#) nella documentazione di IAM.

## eseguire il rehosting (lift and shift)

Vedi [7 R](#).

## sistema little-endian

Un sistema che memorizza per primo il byte meno importante. Vedi anche [endianità](#).

## LLM

Vedi modello [linguistico di grandi dimensioni](#).

## ambienti inferiori

Vedi [ambiente](#).

# M

## machine learning (ML)

Un tipo di intelligenza artificiale che utilizza algoritmi e tecniche per il riconoscimento e l'apprendimento di schemi. Il machine learning analizza e apprende dai dati registrati, come i dati dell'Internet delle cose (IoT), per generare un modello statistico basato su modelli. Per ulteriori informazioni, consulta la sezione [Machine learning](#).

## ramo principale

Vedi [filiale](#).

## malware

Software progettato per compromettere la sicurezza o la privacy del computer. Il malware potrebbe interrompere i sistemi informatici, divulgare informazioni sensibili o ottenere accessi non autorizzati. Esempi di malware includono virus, worm, ransomware, trojan horse, spyware e keylogger.

## servizi gestiti

Servizi AWS per cui AWS gestisce il livello di infrastruttura, il sistema operativo e le piattaforme e si accede agli endpoint per archiviare e recuperare i dati. Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) e Amazon DynamoDB sono esempi di servizi gestiti. Questi sono noti anche come servizi astratti.

## sistema di esecuzione della produzione (MES)

Un sistema software per tracciare, monitorare, documentare e controllare i processi di produzione che convertono le materie prime in prodotti finiti in officina.

## MAP

Vedi [Migration Acceleration Program](#).

## MCP

Vedi [Model Context Protocol](#).

## Model Context Protocol (MCP)

[Un protocollo stateless per la comunicazione tra agenti e strumenti.](#)

## Server MCP

Un servizio che espone uno o più [strumenti](#) tramite il [Model Context](#) Protocol.

## meccanismo

Un processo completo in cui si crea uno strumento, si promuove l'adozione dello strumento e quindi si esaminano i risultati per apportare le modifiche. Un meccanismo è un ciclo che si rafforza e si migliora man mano che funziona. Per ulteriori informazioni, vedete [Creazione di meccanismi](#) nel AWS Well-Architected Framework.

## account membro

Tutti gli account Account AWS diversi dall'account di gestione che fanno parte di un'organizzazione in AWS Organizations. Un account può essere membro di una sola organizzazione alla volta.

## MEH

Vedi [sistema di esecuzione della produzione](#).

## Message Queuing Telemetry Transport (MQTT)

[Un protocollo di comunicazione da macchina a macchina \(M2M\) leggero, basato sul publish/subscribe modello, per dispositivi IoT con risorse limitate.](#)

## microservizio

Un piccolo servizio indipendente che comunica tramite API ben definite ed è in genere di proprietà di piccoli team autonomi. Ad esempio, un sistema assicurativo potrebbe includere microservizi che si riferiscono a funzionalità aziendali, come vendite o marketing, o sottodomini, come acquisti, reclami o analisi. I vantaggi dei microservizi includono agilità, dimensionamento flessibile, facilità di implementazione, codice riutilizzabile e resilienza. [Per ulteriori informazioni, consulta Integrazione dei microservizi utilizzando servizi serverless. AWS](#)

## architettura di microservizi

Un approccio alla creazione di un'applicazione con componenti indipendenti che eseguono ogni processo applicativo come microservizio. Questi microservizi comunicano tramite un'interfaccia ben definita utilizzando API leggere. Ogni microservizio in questa architettura può essere aggiornato, distribuito e dimensionato per soddisfare la richiesta di funzioni specifiche di un'applicazione. Per ulteriori informazioni, vedere [Implementazione](#) dei microservizi su AWS.

## Programma di accelerazione della migrazione (MAP)

Un AWS programma che fornisce consulenza, supporto, formazione e servizi per aiutare le organizzazioni a costruire una solida base operativa per il passaggio al cloud e per contribuire a compensare il costo iniziale delle migrazioni. MAP include una metodologia di migrazione per

eseguire le migrazioni precedenti in modo metodico e un set di strumenti per automatizzare e accelerare gli scenari di migrazione comuni.

## migrazione su larga scala

Il processo di trasferimento della maggior parte del portfolio di applicazioni sul cloud avviene a ondate, con più applicazioni trasferite a una velocità maggiore in ogni ondata. Questa fase utilizza le migliori pratiche e le lezioni apprese nelle fasi precedenti per implementare una fabbrica di migrazione di team, strumenti e processi per semplificare la migrazione dei carichi di lavoro attraverso l'automazione e la distribuzione agile. Questa è la terza fase della [strategia di migrazione AWS](#).

## fabbrica di migrazione

Cross-functional team che semplificano la migrazione dei carichi di lavoro attraverso approcci automatizzati e agili. I team di Migration Factory includono in genere operazioni, analisti e proprietari aziendali, ingegneri addetti alla migrazione, sviluppatori e DevOps professionisti che lavorano nell'ambito degli sprint. Tra il 20% e il 50% di un portfolio di applicazioni aziendali è costituito da schemi ripetuti che possono essere ottimizzati con un approccio di fabbrica. Per ulteriori informazioni, consulta la [discussione sulle fabbriche di migrazione](#) e la [Guida alla fabbrica di migrazione al cloud](#) in questo set di contenuti.

## metadati di migrazione

Le informazioni sull'applicazione e sul server necessarie per completare la migrazione. Ogni modello di migrazione richiede un set diverso di metadati di migrazione. Esempi di metadati di migrazione includono la sottorete, il gruppo di sicurezza e l'account di destinazione. AWS

## modello di migrazione

Un'attività di migrazione ripetibile che descrive in dettaglio la strategia di migrazione, la destinazione della migrazione e l'applicazione o il servizio di migrazione utilizzati. Esempio: riorganizza la migrazione su Amazon EC2 AWS con Application Migration Service.

## Valutazione del portfolio di migrazione (MPA)

Uno strumento online che fornisce informazioni per la convalida del business case per la migrazione a. Cloud AWS MPA offre una valutazione dettagliata del portfolio (dimensionamento corretto dei server, prezzi, confronto del TCO, analisi dei costi di migrazione) e pianificazione della migrazione (analisi e raccolta dei dati delle applicazioni, raggruppamento delle applicazioni, prioritizzazione delle migrazioni e pianificazione delle ondate). [Lo strumento MPA](#) (richiede l'accesso) è disponibile gratuitamente per tutti i AWS consulenti e i consulenti dei partner APN.

## valutazione della preparazione alla migrazione (MRA)

Il processo di acquisizione di informazioni sullo stato di preparazione al cloud di un'organizzazione, l'identificazione dei punti di forza e di debolezza e la creazione di un piano d'azione per colmare le lacune identificate, utilizzando il CAF. AWS Per ulteriori informazioni, consulta la [guida di preparazione alla migrazione](#). MRA è la prima fase della [strategia di migrazione AWS](#).

## strategia di migrazione

L'approccio utilizzato per migrare un carico di lavoro verso. Cloud AWS Per ulteriori informazioni, consulta la voce [7 R](#) in questo glossario e consulta [Mobilita la tua organizzazione per](#) accelerare le migrazioni su larga scala.

## ML

[Vedi machine learning](#).

## modernizzazione

Trasformazione di un'applicazione obsoleta (legacy o monolitica) e della relativa infrastruttura in un sistema agile, elastico e altamente disponibile nel cloud per ridurre i costi, aumentare l'efficienza e sfruttare le innovazioni. Per ulteriori informazioni, vedere [Strategia per la modernizzazione delle applicazioni in](#). Cloud AWS

## valutazione della preparazione alla modernizzazione

Una valutazione che aiuta a determinare la preparazione alla modernizzazione delle applicazioni di un'organizzazione, identifica vantaggi, rischi e dipendenze e determina in che misura l'organizzazione può supportare lo stato futuro di tali applicazioni. Il risultato della valutazione è uno schema dell'architettura di destinazione, una tabella di marcia che descrive in dettaglio le fasi di sviluppo e le tappe fondamentali del processo di modernizzazione e un piano d'azione per colmare le lacune identificate. Per ulteriori informazioni, vedere [Valutazione della preparazione alla modernizzazione per](#) le applicazioni in. Cloud AWS

## applicazioni monolitiche (monoliti)

Applicazioni eseguite come un unico servizio con processi strettamente collegati. Le applicazioni monolitiche presentano diversi inconvenienti. Se una funzionalità dell'applicazione registra un picco di domanda, l'intera architettura deve essere dimensionata. L'aggiunta o il miglioramento delle funzionalità di un'applicazione monolitica diventa inoltre più complessa man mano che la base di codice cresce. Per risolvere questi problemi, puoi utilizzare un'architettura di microservizi. Per ulteriori informazioni, consulta la sezione [Scomposizione dei monoliti in microservizi](#).

## MAPPA

Vedi [Migration Portfolio Assessment](#).

## MQTT

Vedi [Message Queuing Telemetry Transport](#).

## classificazione multiclasse

Un processo che aiuta a generare previsioni per più classi (prevedendo uno o più di due risultati). Ad esempio, un modello di machine learning potrebbe chiedere "Questo prodotto è un libro, un'auto o un telefono?" oppure "Quale categoria di prodotti è più interessante per questo cliente?"

## infrastruttura mutabile

Un modello che aggiorna e modifica l'infrastruttura esistente per i carichi di lavoro di produzione. Per migliorare la coerenza, l'affidabilità e la prevedibilità, il AWS Well-Architected Framework consiglia l'uso di un'infrastruttura [immutabile](#) come best practice.

## O

### OAC

Vedi [Origin Access Control](#).

### QUERCIA

Vedi [Origin Access Identity](#).

### OCM

Vedi [gestione delle modifiche organizzative](#).

## migrazione offline

Un metodo di migrazione in cui il carico di lavoro di origine viene eliminato durante il processo di migrazione. Questo metodo prevede tempi di inattività prolungati e viene in genere utilizzato per carichi di lavoro piccoli e non critici.

## OI

Vedi [l'integrazione delle operazioni](#).

## OLA

Vedi accordo a [livello operativo](#).

## migrazione online

Un metodo di migrazione in cui il carico di lavoro di origine viene copiato sul sistema di destinazione senza essere messo offline. Le applicazioni connesse al carico di lavoro possono continuare a funzionare durante la migrazione. Questo metodo comporta tempi di inattività pari a zero o comunque minimi e viene in genere utilizzato per carichi di lavoro di produzione critici.

## OPC-UA

Vedi [Open Process Communications - Unified Architecture](#).

## Comunicazioni a processo aperto - Architettura unificata () OPC-UA

Un protocollo di comunicazione da macchina a macchina (M2M) per l'automazione industriale. OPC-UA fornisce uno standard di interoperabilità con schemi di crittografia, autenticazione e autorizzazione dei dati.

## accordo a livello operativo (OLA)

Un accordo che chiarisce quali sono gli impegni reciproci tra i gruppi IT funzionali, a supporto di un accordo sul livello di servizio (SLA).

## revisione della prontezza operativa (ORR)

Un elenco di domande e best practice associate che aiutano a comprendere, valutare, prevenire o ridurre la portata degli incidenti e dei possibili guasti. Per ulteriori informazioni, vedere [Operational Readiness Reviews \(ORR\)](#) nel Framework. AWS Well-Architected

## tecnologia operativa (OT)

Sistemi hardware e software che interagiscono con l'ambiente fisico per controllare operazioni, apparecchiature e infrastrutture industriali. Nella produzione, l'integrazione di sistemi OT e di tecnologia dell'informazione (IT) è un obiettivo chiave per le trasformazioni [dell'Industria 4.0](#).

## integrazione delle operazioni (OI)

Il processo di modernizzazione delle operazioni nel cloud, che prevede la pianificazione, l'automazione e l'integrazione della disponibilità. Per ulteriori informazioni, consulta la [guida all'integrazione delle operazioni](#).

## trail organizzativo

Un percorso creato da noi AWS CloudTrail che registra tutti gli eventi di un'organizzazione per tutti Account AWS . AWS Organizations Questo percorso viene creato in ogni Account AWS che

fa parte dell'organizzazione e tiene traccia dell'attività in ogni account. Per ulteriori informazioni, consulta [Creazione di un percorso per un'organizzazione](#) nella CloudTrail documentazione.

#### gestione del cambiamento organizzativo (OCM)

Un framework per la gestione di trasformazioni aziendali importanti e che comportano l'interruzione delle attività dal punto di vista delle persone, della cultura e della leadership. OCM aiuta le organizzazioni a prepararsi e passare a nuovi sistemi e strategie accelerando l'adozione del cambiamento, affrontando i problemi di transizione e promuovendo cambiamenti culturali e organizzativi. Nella strategia di AWS migrazione, questo framework si chiama accelerazione delle persone, a causa della velocità di cambiamento richiesta nei progetti di adozione del cloud. Per ulteriori informazioni, consultare la [Guida OCM](#).

#### controllo dell'accesso all'origine (OAC)

In CloudFront, un'opzione avanzata per limitare l'accesso per proteggere i contenuti di Amazon Simple Storage Service (Amazon S3). OAC supporta in tutto tutti i bucket S3 Regioni AWS, la crittografia lato server con AWS KMS (SSE-KMS) e le richieste dinamiche PUT e dirette al bucket S3. DELETE

#### identità di accesso origine (OAI)

Nel CloudFront, un'opzione per limitare l'accesso per proteggere i tuoi contenuti Amazon S3. Quando usi OAI, CloudFront crea un principale con cui Amazon S3 può autenticarsi. I principali autenticati possono accedere ai contenuti in un bucket S3 solo tramite una distribuzione specifica. CloudFront Vedi anche [OAC](#), che fornisce un controllo degli accessi più granulare e avanzato.

#### ORR

[Vedi la revisione della prontezza operativa.](#)

#### - NON

Vedi la [tecnologia operativa](#).

#### VPC in uscita (egress)

In un'architettura AWS multi-account, un VPC che gestisce le connessioni di rete avviate dall'interno di un'applicazione. Nel documento [Architettura di riferimento per la sicurezza di AWS](#) si consiglia di configurare l'account di rete con VPC in entrata, in uscita e di ispezione per proteggere l'interfaccia bidirezionale tra l'applicazione e Internet in generale.

## P

### limite delle autorizzazioni

Una policy di gestione IAM collegata ai principali IAM per impostare le autorizzazioni massime che l'utente o il ruolo possono avere. Per ulteriori informazioni, consulta [Limiti delle autorizzazioni](#) nella documentazione di IAM.

### informazioni di identificazione personale (PII)

Informazioni che, se visualizzate direttamente o abbinate ad altri dati correlati, possono essere utilizzate per dedurre ragionevolmente l'identità di un individuo. Esempi di informazioni personali includono nomi, indirizzi e informazioni di contatto.

### Informazioni che consentono l'identificazione personale degli utenti

Visualizza le [informazioni di identificazione personale](#).

### playbook

Una serie di passaggi predefiniti che raccolgono il lavoro associato alle migrazioni, come l'erogazione delle funzioni operative principali nel cloud. Un playbook può assumere la forma di script, runbook automatici o un riepilogo dei processi o dei passaggi necessari per gestire un ambiente modernizzato.

### PLC

Vedi [controllore logico programmabile](#).

### PLM

Vedi la gestione [del ciclo di vita del prodotto](#).

### policy

[Un oggetto in grado di definire le autorizzazioni \(vedi politica basata sull'identità\), specificare le condizioni di accesso \(vedi politicabasata sulle risorse\) o definire le autorizzazioni massime per tutti gli account di un'organizzazione in \(vedi politica di controllo dei servizi\). AWS Organizations](#)

### persistenza poliglotta

Scelta indipendente della tecnologia di archiviazione di dati di un microservizio in base ai modelli di accesso ai dati e ad altri requisiti. Se i microservizi utilizzano la stessa tecnologia di archiviazione di dati, possono incontrare problemi di implementazione o registrare prestazioni scadenti. I microservizi vengono implementati più facilmente e ottengono prestazioni e scalabilità migliori se utilizzano l'archivio dati più adatto alle loro esigenze.

## valutazione del portfolio

Un processo di scoperta, analisi e definizione delle priorità del portfolio di applicazioni per pianificare la migrazione. Per ulteriori informazioni, consulta la pagina [Valutazione della preparazione alla migrazione](#).

## predicate

Una condizione di interrogazione che restituisce o, in genere, si trova in una clausola `true`. `false` WHERE

## predicato pushdown

Una tecnica di ottimizzazione delle query del database che filtra i dati della query prima del trasferimento. Ciò riduce la quantità di dati che devono essere recuperati ed elaborati dal database relazionale e migliora le prestazioni delle query.

## controllo preventivo

Un controllo di sicurezza progettato per impedire il verificarsi di un evento. Questi controlli sono la prima linea di difesa per impedire accessi non autorizzati o modifiche indesiderate alla rete. Per ulteriori informazioni, consulta [Controlli preventivi](#) in Implementazione dei controlli di sicurezza in AWS.

## principale

Un'entità in AWS grado di eseguire azioni e accedere alle risorse. Questa entità è in genere un utente root per un Account AWS ruolo IAM o un utente. Per ulteriori informazioni, consulta Principali in [Termini e concetti dei ruoli](#) nella documentazione di IAM.

## privacy fin dalla progettazione

Un approccio ingegneristico dei sistemi che tiene conto della privacy durante l'intero processo di sviluppo.

## zone ospitate private

Un container che contiene informazioni su come si desidera che Amazon Route 53 risponda alle query DNS per un dominio e i relativi sottodomini all'interno di uno o più VPC. Per ulteriori informazioni, consulta [Utilizzo delle zone ospitate private](#) nella documentazione di Route 53.

## controllo proattivo

Un [controllo di sicurezza](#) progettato per impedire l'implementazione di risorse non conformi. Questi controlli analizzano le risorse prima del loro provisioning. Se la risorsa non è conforme al

controllo, non viene fornita. Per ulteriori informazioni, consulta la [guida di riferimento sui controlli](#) nella AWS Control Tower documentazione e consulta Controlli [proattivi in Implementazione dei controlli](#) di sicurezza su. AWS

gestione del ciclo di vita del prodotto (PLM)

La gestione dei dati e dei processi di un prodotto durante l'intero ciclo di vita, dalla progettazione, sviluppo e lancio, attraverso la crescita e la maturità, fino al declino e alla rimozione.

Ambiente di produzione

[Vedi ambiente.](#)

controllore logico programmabile (PLC)

Nella produzione, un computer altamente affidabile e adattabile che monitora le macchine e automatizza i processi di produzione.

concatenamento rapido

Utilizzo dell'output di un prompt [LLM](#) come input per il prompt successivo per generare risposte migliori. Questa tecnica viene utilizzata per suddividere un'attività complessa in sottoattività o per perfezionare o espandere iterativamente una risposta preliminare. Aiuta a migliorare l'accuratezza e la pertinenza delle risposte di un modello e consente risultati più granulari e personalizzati.

pseudonimizzazione

Il processo di sostituzione degli identificatori personali in un set di dati con valori segnaposto. La pseudonimizzazione può aiutare a proteggere la privacy personale. I dati pseudonimizzati sono ancora considerati dati personali.

publish/subscribe (pub/sub)

Un modello che consente comunicazioni asincrone tra microservizi per migliorare la scalabilità e la reattività. Ad esempio, in un [MES](#) basato su microservizi, un microservizio può pubblicare messaggi di eventi su un canale a cui altri microservizi possono abbonarsi. Il sistema può aggiungere nuovi microservizi senza modificare il servizio di pubblicazione.

## Q

Piano di query

Una serie di passaggi, come le istruzioni, utilizzati per accedere ai dati in un sistema di database relazionale SQL.

## regressione del piano di query

Quando un ottimizzatore del servizio di database sceglie un piano non ottimale rispetto a prima di una determinata modifica all'ambiente di database. Questo può essere causato da modifiche a statistiche, vincoli, impostazioni dell'ambiente, associazioni dei parametri di query e aggiornamenti al motore di database.

## R

### Matrice RACI

Vedi [responsabile, responsabile, consultato, informato \(RACI\)](#).

### RAG

Vedi [Retrieval](#) Augmented Generation.

### ransomware

Un software dannoso progettato per bloccare l'accesso a un sistema informatico o ai dati fino a quando non viene effettuato un pagamento.

### Matrice RASCI

Vedi [responsabile, responsabile, consultato, informato \(RACI\)](#).

### RCAC

Vedi controllo dell'[accesso a righe e colonne](#).

### replica di lettura

Una copia di un database utilizzata per scopi di sola lettura. È possibile indirizzare le query alla replica di lettura per ridurre il carico sul database principale.

### riprogettare

Vedi [7 Rs](#).

### obiettivo del punto di ripristino (RPO)

Il periodo di tempo massimo accettabile dall'ultimo punto di ripristino dei dati. Questo determina ciò che si considera una perdita di dati accettabile tra l'ultimo punto di ripristino e l'interruzione del servizio.

## obiettivo del tempo di ripristino (RTO)

Il ritardo massimo accettabile tra l'interruzione del servizio e il ripristino del servizio.

## rifattorizzare

Vedi [7 R.](#)

## Region

Una raccolta di AWS risorse in un'area geografica. Ciascuna Regione AWS è isolata e indipendente dalle altre per fornire tolleranza agli errori, stabilità e resilienza. Per ulteriori informazioni, consulta [Specificare cosa può usare Regioni AWS il tuo account.](#)

## regressione

Una tecnica di ML che prevede un valore numerico. Ad esempio, per risolvere il problema "A che prezzo verrà venduta questa casa?" un modello di ML potrebbe utilizzare un modello di regressione lineare per prevedere il prezzo di vendita di una casa sulla base di dati noti sulla casa (ad esempio, la metratura).

## riospitare

Vedi [7 R.](#)

## rilascio

In un processo di implementazione, l'atto di promuovere modifiche a un ambiente di produzione.

## trasferisco

Vedi [7 Rs.](#)

## ripiattaforma

Vedi [7 Rs.](#)

## riacquisto

Vedi [7 Rs.](#)

## resilienza

La capacità di un'applicazione di resistere o ripristinare le interruzioni. [L'elevata disponibilità e il disaster recovery](#) sono considerazioni comuni quando si pianifica la resilienza in Cloud AWS. [Per ulteriori informazioni, vedere Cloud AWS Resilience.](#)

## policy basata su risorse

Una policy associata a una risorsa, ad esempio un bucket Amazon S3, un endpoint o una chiave di crittografia. Questo tipo di policy specifica a quali principali è consentito l'accesso, le azioni supportate e qualsiasi altra condizione che deve essere soddisfatta.

## matrice di assegnazione di responsabilità (RACI)

Una matrice che definisce i ruoli e le responsabilità di tutte le parti coinvolte nelle attività di migrazione e nelle operazioni cloud. Il nome della matrice deriva dai tipi di responsabilità definiti nella matrice: responsabile (R), responsabile (A), consultato (C) e informato (I). Il tipo di supporto (S) è facoltativo. Se includi il supporto, la matrice viene chiamata matrice RASCI e, se la escludi, viene chiamata matrice RACI.

## controllo reattivo

Un controllo di sicurezza progettato per favorire la correzione di eventi avversi o deviazioni dalla baseline di sicurezza. Per ulteriori informazioni, consulta [Controlli reattivi](#) in Implementazione dei controlli di sicurezza in AWS.

## retain

Vedi [7 R](#).

## andare in pensione

Vedi [7 Rs](#).

## Retrieval Augmented Generation (RAG)

Una tecnologia di [intelligenza artificiale generativa](#) in cui un [LLM](#) fa riferimento a una fonte di dati autorevole esterna alle sue fonti di dati di formazione prima di generare una risposta. Ad esempio, un modello RAG potrebbe eseguire una ricerca semantica nella knowledge base o nei dati personalizzati di un'organizzazione. Per ulteriori informazioni, consulta [Cos'è il RAG](#).

## rotazione

Processo di aggiornamento periodico di un [segreto](#) per rendere più difficile l'accesso alle credenziali da parte di un utente malintenzionato.

## controllo dell'accesso a righe e colonne (RCAC)

L'uso di espressioni SQL di base e flessibili con regole di accesso definite. RCAC è costituito da autorizzazioni di riga e maschere di colonna.

## RPO

Vedi [obiettivo del punto di ripristino](#).

## VERSO

Vedi [obiettivo del tempo di ripristino](#).

## runbook

Un insieme di procedure manuali o automatizzate necessarie per eseguire un'attività specifica. In genere sono progettati per semplificare operazioni o procedure ripetitive con tassi di errore elevati.

## S

### SAML 2.0

Uno standard aperto utilizzato da molti provider di identità (IdPs). Questa funzionalità abilita il single sign-on (SSO) federato, in modo che gli utenti possano accedere Console di gestione AWS o chiamare le operazioni AWS API senza che tu debba creare un utente in IAM per tutti i membri dell'organizzazione. Per ulteriori informazioni sulla federazione basata su SAML 2.0, consulta [Informazioni sulla federazione basata su SAML 2.0](#) nella documentazione di IAM.

### SCADA

Vedi [controllo di supervisione e acquisizione dati](#).

### SCP

Vedi la [politica di controllo del servizio](#).

### Secret

In Gestione dei segreti AWS, informazioni riservate o riservate, come una password o le credenziali utente, archiviate in forma crittografata. È costituito dal valore segreto e dai relativi metadati. Il valore segreto può essere binario, una stringa singola o più stringhe. Per ulteriori informazioni, consulta [Cosa c'è in un segreto di Secrets Manager?](#) nella documentazione di Secrets Manager.

### sicurezza fin dalla progettazione

Un approccio di ingegneria dei sistemi che tiene conto della sicurezza durante l'intero processo di sviluppo.

## controllo di sicurezza

Un guardrail tecnico o amministrativo che impedisce, rileva o riduce la capacità di un autore di minacce di sfruttare una vulnerabilità di sicurezza. [Esistono quattro tipi principali di controlli di sicurezza: preventivi, investigativi, reattivi e proattivi.](#)

## rafforzamento della sicurezza

Il processo di riduzione della superficie di attacco per renderla più resistente agli attacchi. Può includere azioni come la rimozione di risorse che non sono più necessarie, l'implementazione di best practice di sicurezza che prevedono la concessione del privilegio minimo o la disattivazione di funzionalità non necessarie nei file di configurazione.

## sistema di gestione delle informazioni e degli eventi di sicurezza (SIEM)

Strumenti e servizi che combinano sistemi di gestione delle informazioni di sicurezza (SIM) e sistemi di gestione degli eventi di sicurezza (SEM). Un sistema SIEM raccoglie, monitora e analizza i dati da server, reti, dispositivi e altre fonti per rilevare minacce e violazioni della sicurezza e generare avvisi.

## automazione della risposta alla sicurezza

Un'azione predefinita e programmata progettata per rispondere o porre rimedio automaticamente a un evento di sicurezza. Queste automazioni fungono da controlli di sicurezza [investigativi](#) o [reattivi](#) che aiutano a implementare le migliori pratiche di sicurezza. AWS Esempi di azioni di risposta automatizzate includono la modifica di un gruppo di sicurezza VPC, l'applicazione di patch a un'istanza Amazon EC2 o la rotazione delle credenziali.

## Crittografia lato server

Crittografia dei dati a destinazione, da parte di chi li riceve. Servizio AWS

## Policy di controllo dei servizi (SCP)

Una policy che fornisce il controllo centralizzato sulle autorizzazioni per tutti gli account di un'organizzazione in AWS Organizations. Le SCP definiscono i guardrail o fissano i limiti alle azioni che un amministratore può delegare a utenti o ruoli. Puoi utilizzare le SCP come elenchi consentiti o elenchi di rifiuto, per specificare quali servizi o azioni sono consentiti o proibiti. Per ulteriori informazioni, consulta [le politiche di controllo del servizio](#) nella AWS Organizations documentazione.

## endpoint del servizio

L'URL del punto di ingresso per un Servizio AWS. Puoi utilizzare l'endpoint per connetterti a livello di programmazione al servizio di destinazione. Per ulteriori informazioni, consulta [Endpoint del Servizio AWS](#) nei Riferimenti generali di AWS.

## accordo sul livello di servizio (SLA)

Un accordo che chiarisce ciò che un team IT promette di offrire ai propri clienti, ad esempio l'operatività e le prestazioni del servizio.

## indicatore del livello di servizio (SLI)

Misurazione di un aspetto prestazionale di un servizio, ad esempio il tasso di errore, la disponibilità o la velocità effettiva.

## obiettivo a livello di servizio (SLO)

[Una metrica target che rappresenta lo stato di un servizio, misurato da un indicatore del livello di servizio.](#)

## Modello di responsabilità condivisa

Un modello che descrive la responsabilità condivisa AWS per la sicurezza e la conformità del cloud. AWS è responsabile della sicurezza del cloud, mentre tu sei responsabile della sicurezza nel cloud. Per ulteriori informazioni, consulta [Modello di responsabilità condivisa](#).

## Shadow AI

Applicazioni di [intelligenza artificiale](#) non autorizzate create o utilizzate al di fuori dei canali regolamentati all'interno di un'organizzazione.

## SIEM

Vedi il [sistema di gestione delle informazioni e degli eventi sulla sicurezza](#).

## punto di errore singolo (SPOF)

Un guasto in un singolo componente critico di un'applicazione che può disturbare il sistema.

## SLAM

Vedi il contratto sul [livello di servizio](#).

## SLI

Vedi l'indicatore del [livello di servizio](#).

## LENTA

Vedi obiettivo del [livello di servizio](#).

### modello split-and-seed

Un modello per dimensionare e accelerare i progetti di modernizzazione. Man mano che vengono definite nuove funzionalità e versioni dei prodotti, il team principale si divide per creare nuovi team di prodotto. Questo aiuta a dimensionare le capacità e i servizi dell'organizzazione, migliora la produttività degli sviluppatori e supporta una rapida innovazione. Per ulteriori informazioni, vedere [Approccio graduale alla modernizzazione delle applicazioni in](#). Cloud AWS

## SPOF

Vedi [punto di errore singolo](#).

### schema a stella

Una struttura organizzativa di database che utilizza un'unica tabella dei fatti di grandi dimensioni per archiviare i dati transazionali o misurati e utilizza una o più tabelle dimensionali più piccole per memorizzare gli attributi dei dati. Questa struttura è progettata per l'uso in un [data warehouse](#) o per scopi di business intelligence.

### modello del fico strangolatore

Un approccio alla modernizzazione dei sistemi monolitici mediante la riscrittura e la sostituzione incrementali delle funzionalità del sistema fino alla disattivazione del sistema legacy. Questo modello utilizza l'analogia di una pianta di fico che cresce fino a diventare un albero robusto e alla fine annienta e sostituisce il suo ospite. Il modello è stato [introdotto da Martin Fowler](#) come metodo per gestire il rischio durante la riscrittura di sistemi monolitici. Per un esempio di come applicare questo modello, consulta [Modernizzare i servizi Web Microsoft ASP.NET \(ASMX\) legacy in modo incrementale utilizzando contenitori e Amazon API Gateway](#).

### sottorete

Un intervallo di indirizzi IP nel VPC. Una sottorete deve risiedere in una singola zona di disponibilità.

### controllo di supervisione e acquisizione dati (SCADA)

Nella produzione, un sistema che utilizza hardware e software per monitorare gli asset fisici e le operazioni di produzione.

### crittografia simmetrica

Un algoritmo di crittografia che utilizza la stessa chiave per crittografare e decrittografare i dati.

## test sintetici

Test di un sistema in modo da simulare le interazioni degli utenti per rilevare potenziali problemi o monitorare le prestazioni. Puoi usare [Amazon CloudWatch Synthetics](#) per creare questi test.

## prompt di sistema

Una tecnica per fornire contesto, istruzioni o linee guida a un [LLM](#) per indirizzarne il comportamento. I prompt di sistema aiutano a impostare il contesto e stabilire regole per le interazioni con gli utenti.

# T

## tag

Key-value coppie che fungono da metadati per l'organizzazione delle risorse. AWS Con i tag è possibile a gestire, identificare, organizzare, cercare e filtrare le risorse. Per ulteriori informazioni, consulta [Tagging delle risorse AWS](#).

## variabile di destinazione

Il valore che stai cercando di prevedere nel machine learning supervisionato. Questo è indicato anche come variabile di risultato. Ad esempio, in un ambiente di produzione la variabile di destinazione potrebbe essere un difetto del prodotto.

## elenco di attività

Uno strumento che viene utilizzato per tenere traccia dei progressi tramite un runbook. Un elenco di attività contiene una panoramica del runbook e un elenco di attività generali da completare. Per ogni attività generale, include la quantità stimata di tempo richiesta, il proprietario e lo stato di avanzamento.

## ambiente di test

Vedi [ambiente](#).

## training

Fornire dati da cui trarre ispirazione dal modello di machine learning. I dati di training devono contenere la risposta corretta. L'algoritmo di apprendimento trova nei dati di addestramento i pattern che mappano gli attributi dei dati di input al target (la risposta che si desidera prevedere). Produce un modello di ML che acquisisce questi modelli. Puoi quindi utilizzare il modello di ML per creare previsioni su nuovi dati di cui non si conosce il target.

## strumento

Una funzione o API che un [agente](#) può richiamare per eseguire operazioni in sistemi esterni.

## Transit Gateway

Un hub di transito di rete che è possibile utilizzare per collegare i VPC e le reti on-premise. Per ulteriori informazioni, consulta [Cos'è un gateway di transito](#) nella AWS Transit Gateway documentazione.

## flusso di lavoro basato su trunk

Un approccio in cui gli sviluppatori creano e testano le funzionalità localmente in un ramo di funzionalità e quindi uniscono tali modifiche al ramo principale. Il ramo principale viene quindi integrato negli ambienti di sviluppo, produzione e produzione, in sequenza.

## Accesso attendibile

Concessione delle autorizzazioni a un servizio specificato dall'utente per eseguire attività all'interno dell'organizzazione AWS Organizations e nei suoi account per conto dell'utente. Il servizio attendibile crea un ruolo collegato al servizio in ogni account, quando tale ruolo è necessario, per eseguire attività di gestione per conto dell'utente. Per ulteriori informazioni, consulta [Utilizzo AWS Organizations con altri AWS servizi](#) nella AWS Organizations documentazione.

## regolazione

Modificare alcuni aspetti del processo di training per migliorare la precisione del modello di ML. Ad esempio, puoi addestrare il modello di ML generando un set di etichette, aggiungendo etichette e quindi ripetendo questi passaggi più volte con impostazioni diverse per ottimizzare il modello.

## team da due pizze

Una piccola DevOps squadra che puoi sfamare con due pizze. Un team composto da due persone garantisce la migliore opportunità possibile di collaborazione nello sviluppo del software.

# U

## incertezza

Un concetto che si riferisce a informazioni imprecise, incomplete o sconosciute che possono minare l'affidabilità dei modelli di machine learning predittivi. Esistono due tipi di incertezza:

l'incertezza epistemica, che è causata da dati limitati e incompleti, mentre l'incertezza aleatoria è causata dal rumore e dalla casualità insiti nei dati.

### compiti indifferenziati

Conosciuto anche come sollevamento di carichi pesanti, è un lavoro necessario per creare e far funzionare un'applicazione, ma che non apporta valore diretto all'utente finale né offre vantaggi competitivi. Esempi di attività indifferenziate includono l'approvvigionamento, la manutenzione e la pianificazione della capacità.

### ambienti superiori

[Vedi ambiente.](#)

## V

### vacuum

Un'operazione di manutenzione del database che prevede la pulizia dopo aggiornamenti incrementali per recuperare lo spazio di archiviazione e migliorare le prestazioni.

### controllo delle versioni

Processi e strumenti che tengono traccia delle modifiche, ad esempio le modifiche al codice di origine in un repository.

### Peering VPC

Una connessione tra due VPC che consente di instradare il traffico tramite indirizzi IP privati. Per ulteriori informazioni, consulta [Che cos'è il peering VPC?](#) nella documentazione di Amazon VPC.

### vulnerabilità

Un difetto software o hardware che compromette la sicurezza del sistema.

## W

### cache calda

Una cache del buffer che contiene dati correnti e pertinenti a cui si accede frequentemente. L'istanza di database può leggere dalla cache del buffer, il che richiede meno tempo rispetto alla lettura dalla memoria dal disco principale.

## dati caldi

Dati a cui si accede raramente. Quando si eseguono interrogazioni di questo tipo di dati, in genere sono accettabili interrogazioni moderatamente lente.

## funzione finestra

Una funzione SQL che esegue un calcolo su un gruppo di righe che si riferiscono in qualche modo al record corrente. Le funzioni della finestra sono utili per l'elaborazione di attività, come il calcolo di una media mobile o l'accesso al valore delle righe in base alla posizione relativa della riga corrente.

## Carico di lavoro

Una raccolta di risorse e codice che fornisce valore aziendale, ad esempio un'applicazione rivolta ai clienti o un processo back-end.

## flusso di lavoro

Gruppi funzionali in un progetto di migrazione responsabili di una serie specifica di attività. Ogni flusso di lavoro è indipendente ma supporta gli altri flussi di lavoro del progetto. Ad esempio, il flusso di lavoro del portfolio è responsabile della definizione delle priorità delle applicazioni, della pianificazione delle ondate e della raccolta dei metadati di migrazione. Il flusso di lavoro del portfolio fornisce queste risorse al flusso di lavoro di migrazione, che quindi migra i server e le applicazioni.

## VERME

Vedi [scrivere una volta, leggere molti](#).

## WQF

Vedi [AWS Workload Qualification Framework](#).

## scrivi una volta, leggi molte (WORM)

Un modello di storage che scrive i dati una sola volta e ne impedisce l'eliminazione o la modifica. Gli utenti autorizzati possono leggere i dati tutte le volte che è necessario, ma non possono modificarli. Questa infrastruttura di archiviazione dei dati è considerata [immutabile](#).

## Z

### exploit zero-day

[Un attacco, in genere malware, che sfrutta una vulnerabilità zero-day.](#)

## vulnerabilità zero-day

Un difetto o una vulnerabilità assoluta in un sistema di produzione. Gli autori delle minacce possono utilizzare questo tipo di vulnerabilità per attaccare il sistema. Gli sviluppatori vengono spesso a conoscenza della vulnerabilità causata dall'attacco.

## prompt zero-shot

Fornire a un [LLM](#) le istruzioni per eseguire un'attività ma non esempi (immagini) che possano aiutarla. Il LLM deve utilizzare le sue conoscenze pre-addestrate per gestire l'attività. L'efficacia del prompt zero-shot dipende dalla complessità dell'attività e dalla qualità del prompt. [Vedi anche few-shot prompting.](#)

## applicazione zombie

Un'applicazione che prevede un utilizzo CPU e memoria inferiore al 5%. In un progetto di migrazione, è normale ritirare queste applicazioni.

Le traduzioni sono generate tramite traduzione automatica. In caso di conflitto tra il contenuto di una traduzione e la versione originale in Inglese, quest'ultima prevarrà.