



Modellazione dei dati con Amazon DynamoDB

# AWS Guida prescrittiva



# AWS Guida prescrittiva: Modellazione dei dati con Amazon DynamoDB

Copyright © 2024 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 delle rispettive aziende, che possono o meno essere associate, collegate o sponsorizzate da Amazon.

---

# Table of Contents

Introduzione .....	1
Flusso del processo .....	2
Matrice RACI .....	2
Fasi del processo .....	5
Fase 1. Identifica i casi d'uso e il modello logico di dati .....	5
Obiettivi .....	5
Processo .....	5
Strumenti e risorse .....	6
TRACCIA .....	6
Output .....	6
Fase 2. Crea una stima preliminare dei costi .....	6
Obiettivo .....	6
Processo .....	6
Strumenti e risorse .....	7
TRACI .....	7
Output .....	7
Fase 3. Identifica i tuoi modelli di accesso ai dati .....	7
Obiettivo .....	7
Processo .....	7
Strumenti e risorse .....	8
TRACI .....	8
Output .....	9
Esempio .....	9
Fase 4. Identifica i requisiti tecnici .....	9
Obiettivo .....	9
Processo .....	9
Strumenti e risorse .....	10
PISTA .....	10
Output .....	10
Fase 5. Creare il modello di dati DynamoDB .....	10
Obiettivo .....	10
Processo .....	10
Strumenti e risorse .....	12
TRACI .....	12

Output .....	12
Esempio .....	12
Fase 6. Crea le interrogazioni sui dati .....	13
Obiettivo .....	13
Processo .....	13
Strumenti e risorse .....	14
TRACI .....	14
Output .....	14
Esempi .....	14
Fase 7. Convalida il modello di dati .....	14
Obiettivo .....	14
Processo .....	15
Strumenti e risorse .....	15
TRACI .....	15
Output .....	15
Fase 8. Rivedi la stima dei costi .....	16
Obiettivi .....	16
Processo .....	16
Strumenti e risorse .....	16
TRACI .....	16
Output .....	17
Fase 9. Implementa il modello di dati .....	17
Obiettivo .....	17
Processo .....	17
Strumenti e risorse .....	17
TRACI .....	17
Output .....	17
Esempio .....	18
Modelli .....	19
Modello di valutazione dei requisiti aziendali .....	19
Modello di valutazione dei requisiti tecnici .....	22
Modello di modelli di accesso .....	27
Modello .....	28
Best practice .....	32
Modellazione gerarchica dei dati .....	33
Fase 1: Identifica i casi d'uso e il modello logico di dati .....	33

Fase 2: Creare una stima preliminare dei costi .....	35
Fase 3: Identifica i tuoi modelli di accesso ai dati .....	36
Fase 4: Identificare i requisiti tecnici .....	37
Fase 5: Creare un modello di dati DynamoDB .....	37
Archiviazione dei componenti nella tabella .....	38
L'indice GSI1 .....	39
L'indice GSI2 .....	40
Fase 6: Creare interrogazioni di dati .....	41
Fase 7: Convalida del modello di dati .....	44
Fase 8: Rivedere la stima dei costi .....	45
Obiettivi .....	46
Processo .....	46
Fase 9: Implementazione del modello di dati .....	46
Risorse aggiuntive .....	48
Collaboratori .....	50
Cronologia dei documenti .....	51
Glossario .....	52
# .....	52
A .....	53
B .....	56
C .....	58
D .....	61
E .....	65
F .....	67
G .....	68
H .....	69
I .....	70
L .....	73
M .....	74
O .....	78
P .....	81
Q .....	83
R .....	84
S .....	87
T .....	90
U .....	92

---

V .....	92
W .....	93
Z .....	94
.....	<b>xcv</b>

# Modellazione dei dati con Amazon DynamoDB

## Processo, modelli e best practice

Amazon Web Services (AWS)

Dicembre 2023 ([cronologia dei documenti](#))

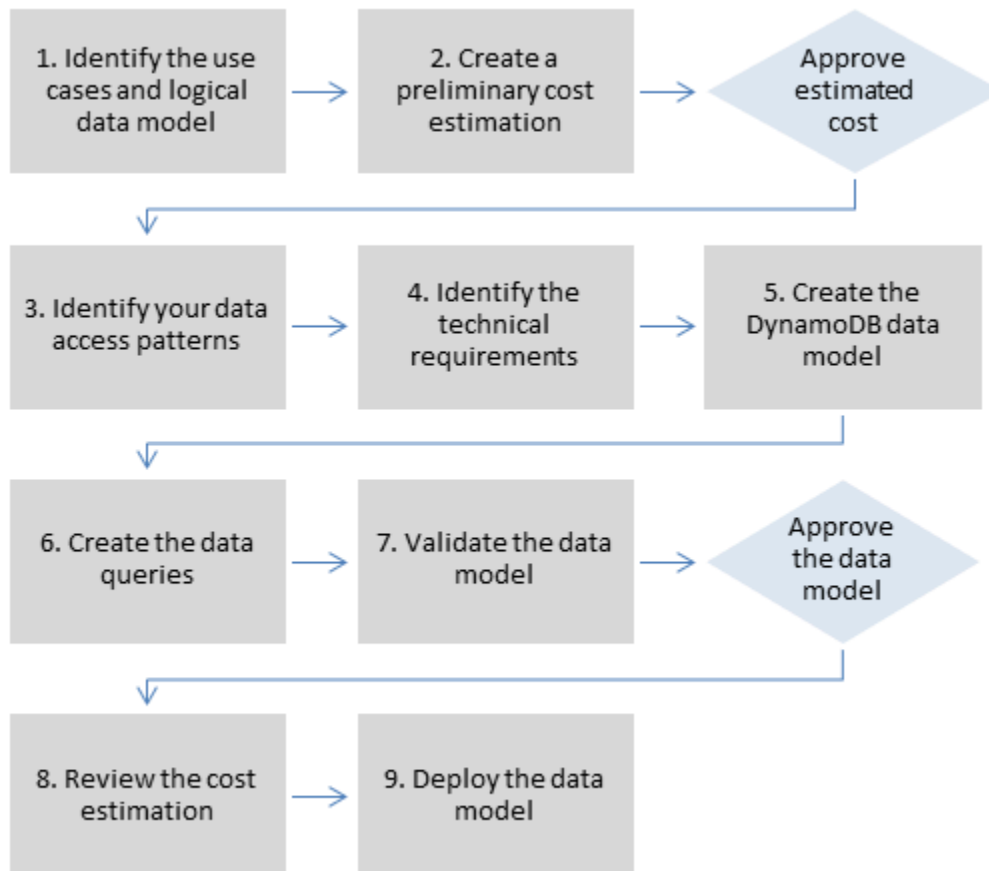
I database NoSQL forniscono schemi flessibili per la creazione di applicazioni moderne. Sono ampiamente riconosciuti per la loro facilità di sviluppo, funzionalità e prestazioni su larga scala. Amazon DynamoDB offre prestazioni veloci e prevedibili con una scalabilità perfetta per i database NoSQL nel cloud Amazon Web Services (AWS). In quanto servizio di database completamente gestito, DynamoDB consente di alleggerire gli oneri amministrativi legati al funzionamento e alla scalabilità di un database distribuito. Non devi preoccuparti del provisioning, dell'installazione e della configurazione dell'hardware, della replica, dell'applicazione di patch software o della scalabilità del cluster.

La progettazione dello schema NoSQL richiede un approccio diverso rispetto alla progettazione tradizionale del sistema di gestione di database relazionali (RDBMS). Il modello di dati RDBMS si concentra sulla struttura dei dati e sulle sue relazioni con altri dati. La modellazione dei dati NoSQL si concentra sui modelli di accesso o sul modo in cui l'applicazione consumerà i dati, quindi archivia i dati in modo da supportare operazioni di interrogazione semplici. Per un RDBMS come Microsoft SQL Server o IBM Db2, puoi creare un modello di dati normalizzato senza pensare troppo ai modelli di accesso. È possibile estendere il modello di dati per supportare modelli e query in un secondo momento.

Questa guida presenta un processo di modellazione dei dati per l'utilizzo di DynamoDB che fornisce requisiti funzionali, prestazioni e costi effettivi. La guida è rivolta agli ingegneri di database che intendono utilizzare DynamoDB come database operativo per le applicazioni su cui sono in esecuzione. AWS Professional Services ha utilizzato la procedura consigliata per aiutare le aziende aziendali con la modellazione dei dati DynamoDB per diversi casi d'uso e carichi di lavoro.

# Flusso del processo di modellazione dei dati

Per la modellazione dei dati con Amazon DynamoDB, consigliamo il seguente processo. I passaggi sono descritti in dettaglio [più avanti in questa guida](#).



## Matrice RACI

Alcune organizzazioni utilizzano una matrice di assegnazione delle responsabilità (nota anche come matrice RACI) per descrivere i vari ruoli coinvolti in uno specifico progetto o processo aziendale.

Questa guida presenta una matrice RACI consigliata che potrebbe aiutare l'organizzazione a identificare le persone giuste e le responsabilità giuste per il processo di modellazione dei dati DynamoDB. Per ogni fase del processo, elenca le parti interessate e il loro coinvolgimento:

- R — responsabile del completamento del passaggio
- A — responsabile dell'approvazione e della firma del lavoro



- C — consultato per fornire input per un'attività
- I — informato sui progressi, ma non direttamente coinvolto nell'attività

A seconda della struttura dell'organizzazione e del team di progetto, i ruoli nella seguente matrice RACI possono essere svolti dallo stesso stakeholder. In alcune situazioni, le parti interessate sono sia responsabili che responsabili di fasi specifiche. Ad esempio, gli ingegneri del database possono essere responsabili sia della creazione che dell'approvazione del modello di dati, poiché questa è la loro area di dominio.

Fase del processo	Utente aziendale	Business analyst	Architetto di soluzioni	Ingegnere di database	Sviluppatori di applicazioni	DevOps ingegnere
1. Identificare i casi d'uso e il modello logico dei dati	C	R/A	I	R		
2. Creare una stima preliminare dei costi	C	A	I	R		
3. Identificare i tuoi modelli di accesso ai dati	C	A	I	R		
4. Identificare i requisiti tecnici	C	C	A	R		
5. Creare il modello	I	I	I	R/A		

Fase del processo	Utente aziendale	Business analyst	Architetto di soluzioni	Ingegnere di database	Sviluppatori di applicazioni	DevOps ingegnere
di dati DynamoDB						
6. Crea le interrogazioni sui dati	I	I	I	R/A	R	
7. Convalida il modello di dati	A	R	I	C		
8. Rivedi la stima dei costi	C	A	I	R		
9. Implementa il modello di dati DynamoDB	I	I	C	C		R/A

# Fasi del processo di modellazione dei dati

Questa sezione descrive in dettaglio ogni fase del processo di modellazione dei dati consigliato per Amazon DynamoDB.

## Argomenti

- [Fase 1. Identifica i casi d'uso e il modello logico di dati](#)
- [Fase 2. Crea una stima preliminare dei costi](#)
- [Fase 3. Identifica i tuoi modelli di accesso ai dati](#)
- [Fase 4. Identifica i requisiti tecnici](#)
- [Fase 5. Creare il modello di dati DynamoDB](#)
- [Fase 6. Crea le interrogazioni sui dati](#)
- [Fase 7. Convalida il modello di dati](#)
- [Fase 8. Rivedi la stima dei costi](#)
- [Fase 9. Implementa il modello di dati](#)

## Fase 1. Identifica i casi d'uso e il modello logico di dati

### Obiettivi

- Raccogli le esigenze aziendali e i casi d'uso che richiedono un database NoSQL.
- Definisci il modello logico di dati utilizzando un diagramma entità-relazione (ER).

### Processo

- Gli analisti aziendali intervistano gli utenti aziendali per identificare i casi d'uso e i risultati attesi.
- L'ingegnere del database crea il modello concettuale dei dati.
- L'ingegnere del database crea il modello logico di dati.
- L'ingegnere del database raccoglie informazioni sulla dimensione degli elementi, sul volume dei dati e sulla velocità di lettura e scrittura prevista.

## Strumenti e risorse

- Valutazione dei requisiti aziendali (vedi [modello](#))
- Matrice dei modelli di accesso (vedi [modello](#))
- Il tuo strumento preferito per creare diagrammi

## TRACCIA

Utente aziendale	Business analyst	Architetto di soluzioni	Ingegnere di database	Sviluppatore di applicazioni	DevOps ingegnere
C	R/A	I	R		

## Output

- Casi d'uso e requisiti aziendali documentati
- Modello logico di dati (diagramma ER)

## Fase 2. Crea una stima preliminare dei costi

### Obiettivo

- Sviluppa una stima preliminare dei costi per DynamoDB.

### Processo

- L'ingegnere del database crea l'analisi iniziale dei costi utilizzando le informazioni disponibili e gli esempi presentati nella pagina dei prezzi di [DynamoDB](#).
  - [Crea una stima dei costi per la capacità su richiesta \(vedi esempio\)](#).
  - Crea una stima dei costi per la capacità assegnata (vedi [esempio](#)).
    - Per il modello di capacità assegnata, ottieni il costo stimato dal calcolatore e applica lo sconto per la capacità riservata.

- Confrontate i costi stimati dei due modelli di capacità.
- Crea una stima per tutti gli ambienti (Dev, Prod, QA).
- L'analista aziendale esamina e approva o rifiuta la stima preliminare dei costi.

## Strumenti e risorse

- [AWS Pricing Calculator](#)

## TRACI

Utente aziendale	Business analyst	Architetto di soluzioni	Ingegnere di database	Sviluppatore di applicazioni	DevOps ingegnere
C	A	I	R		

## Output

- stima preliminare dei costi

## Fase 3. Identifica i tuoi modelli di accesso ai dati

I modelli di accesso o i modelli di interrogazione definiscono il modo in cui gli utenti e il sistema accedono ai dati per soddisfare le esigenze aziendali.

## Obiettivo

- Documenta i modelli di accesso ai dati.

## Processo

- L'ingegnere di database e l'analista aziendale intervistano gli utenti finali per identificare come verranno interrogati i dati utilizzando il modello di matrice dei modelli di accesso ai dati.

- Per quanto riguarda le nuove applicazioni, esaminano le storie degli utenti su attività e obiettivi. Documentano i casi d'uso e analizzano i modelli di accesso richiesti dai casi d'uso.
- Per le applicazioni esistenti, analizzano i registri delle interrogazioni per scoprire in che modo le persone utilizzano attualmente il sistema e per identificare i principali modelli di accesso.
- L'ingegnere del database identifica le seguenti proprietà dei modelli di accesso:
  - Dimensione dei dati: sapere quanti dati verranno archiviati e richiesti contemporaneamente aiuta a determinare il modo più efficace per partizionare i dati (vedi il [post del blog](#)).
  - Forma dei dati: invece di modificare i dati quando viene elaborata una query (come nei sistemi RDBMS), un database NoSQL organizza i dati in modo che la loro forma nel database corrisponda al soggetto della query. Questo è un fattore chiave nell'aumentare la velocità e la scalabilità.
  - Velocità dei dati: DynamoDB si dimensiona aumentando il numero di partizioni fisiche disponibili per eseguire le query e distribuendo efficacemente i dati nelle partizioni. Conoscere in anticipo i picchi di carico delle query può aiutare a determinare come partizionare i dati per utilizzare al meglio la capacità di I/O.
- L'utente aziendale dà priorità agli schemi di accesso o di query.
  - Le query prioritarie sono in genere le query più utilizzate o più pertinenti. È inoltre importante identificare le query che richiedono una latenza di risposta inferiore.

## Strumenti e risorse

- Matrice dei modelli di accesso (vedi [modello](#))
- [Scelta della chiave di partizione DynamoDB giusta \(blog AWS Database\)](#)
- [Progettazione NoSQL per DynamoDB \(documentazione DynamoDB\)](#)

## TRACI

Utente aziendale	Business analyst	Architetto di soluzioni	Ingegnere di database	Sviluppatore di applicazioni	DevOps ingegnere
C	A	I	R		

## Output

- matrice dei modelli di accesso ai dati

## Esempio

Schema di accesso	Priority (Priorità)	Leggi o scrivi	Descrizione	Tipo (elemento singolo, elementi multipli o tutti)	Attributo chiave	Filtri	Ordinamento dei risultati
Crea un profilo utente	Elevata	Scrittura	L'utente crea un nuovo profilo	Elemento singolo	Username	N/D	N/D
Aggiorna il profilo utente	Media	Scrittura	L'utente aggiorna il proprio profilo	Elemento singolo	Username	Nome utente = utente corrente	N/D

## Fase 4. Identifica i requisiti tecnici

### Obiettivo

- Raccogli i requisiti tecnici per il database DynamoDB.

### Processo

- Gli analisti aziendali intervistano l'utente e il DevOps team aziendali per raccogliere i requisiti tecnici utilizzando il questionario di valutazione.

## Strumenti e risorse

- Valutazione dei requisiti tecnici (vedi [esempio di questionario](#))

## PISTA

Utente aziendale	Business analyst	Architetto di soluzioni	Ingegnere di database	Sviluppatore di applicazioni	DevOps ingegnere
C	C	A	R		

## Output

- Documento sui requisiti tecnici

## Fase 5. Creare il modello di dati DynamoDB

### Obiettivo

- Crea il modello di dati DynamoDB.

### Processo

- L'ingegnere del database identifica quante tabelle saranno necessarie per ogni caso d'uso. Consigliamo di mantenere il minor numero possibile di tabelle in un'applicazione DynamoDB.
- In base ai modelli di accesso più comuni, identifica la chiave primaria che può essere di due tipi: una chiave primaria con una chiave di partizione che identifica i dati o una chiave primaria con una chiave di partizione e una chiave di ordinamento. Una chiave di ordinamento è una chiave secondaria per raggruppare e organizzare i dati in modo che possano essere interrogati all'interno di una partizione in modo efficiente. [È possibile utilizzare le chiavi di ordinamento per definire relazioni gerarchiche nei dati su cui interrogare a qualsiasi livello della gerarchia \(vedi il post del blog\).](#)
  - Progettazione delle chiavi di partizione



- Definisci la chiave di partizione e valuta la sua distribuzione.
- Identifica la necessità di uno [sharding di scrittura](#) per distribuire i carichi di lavoro in modo uniforme.
- Progettazione della chiave di ordinamento
  - Identifica la chiave di ordinamento.
  - Identifica la necessità di una chiave di ordinamento composita.
  - Identifica la necessità del controllo della versione.
- In base ai modelli di accesso, identifica gli indici secondari per soddisfare i requisiti di interrogazione.
  - Identifica la necessità di [indici secondari locali \(LSI\)](#). Si tratta di indici che hanno la stessa chiave di partizione della tabella di base, ma una chiave di ordinamento diversa.
    - Per le tabelle con LSI, esiste un limite di dimensione di 10 GB per valore della chiave di partizione. Una tabella con LSI può memorizzare un numero qualsiasi di elementi, purché la dimensione totale di ogni valore della chiave di partizione non superi i 10 GB.
  - Identifica la necessità di [indici secondari globali \(GSI\)](#). [Si tratta di indici che hanno una chiave di partizione e una chiave di ordinamento che possono essere diverse da quelle della tabella di base \(vedi il post del blog\)](#).
  - Definite le proiezioni dell'indice. Considera la proiezione di un numero minore di attributi per ridurre la dimensione delle voci scritte nell'indice. In questo passaggio, è necessario determinare se si desidera utilizzare quanto segue:
    - [Indici sparsi](#)
    - [Interrogazioni di aggregazione materializzate](#)
    - [Sovraccarico del GSI](#)
    - [Sharding GSI](#)
    - [Una replica finalmente coerente utilizzando GSI](#)
- L'ingegnere del database determina se i dati includeranno elementi di grandi dimensioni. In tal caso, progettano la soluzione [utilizzando la compressione o archiviando i dati](#) in Amazon Simple Storage Service (Amazon S3).
- L'ingegnere del database determina se saranno necessari i dati delle serie temporali. In tal caso, utilizzano il [modello di progettazione delle serie temporali](#) per modellare i dati.
- L'ingegnere del database determina se il modello ER include many-to-many relazioni. In tal caso, utilizzano un modello di [progettazione a elenco di adiacenze per modellare](#) i dati.

## Strumenti e risorse

- [NoSQL Workbench per Amazon DynamoDB: fornisce funzionalità di modellazione, visualizzazione dei dati e sviluppo e test di query per aiutarti a progettare il tuo database DynamoDB](#)
- [Progettazione NoSQL per DynamoDB \(documentazione DynamoDB\)](#)
- [Scelta della chiave di partizione DynamoDB giusta \(blog AWS Database\)](#)
- [Le migliori pratiche per l'utilizzo degli indici secondari in DynamoDB \(documentazione DynamoDB\)](#)
- [Come progettare indici secondari globali di Amazon DynamoDB \(blog AWS Database\)](#)

## TRACI

Utente aziendale	Business analyst	Architetto di soluzioni	Ingegnere di database	Sviluppatore di applicazioni	DevOps ingegnere
			R/A		

## Output

- Schema di tabella DynamoDB che soddisfa i tuoi modelli e requisiti di accesso

## Esempio

La schermata seguente mostra NoSQL Workbench.

RetailDatabase		GSI: BundleProducts-VendorProducts-CustomerOrdersByOrderId		GSI: VendorOrdersByStatusDate-ProductInventory		GSI: ProductCatalog-CustomerOrdersByProduct	
Primary Key		Attributes					
Partition Key: pk	Sort Key: sk						
P1	B1	GS11-PK	GS11-SK	name	desc		
		B1	P1	The Tiki Bundle	Everything you need for an island theme party.		
P4	B2	GS11-PK	GS11-SK	name	desc		
		B2	P4	Tiki Bar Set	Be the Mai Tai master with your very own Tiki Bar.		
P2	B1	name	desc	qty	GS11-PK	GS11-SK	location
		Tiki Torch	Bamboo tiki torch, 4 ft	6	B1	P2	W1-A9-S10-B52
	B2	name	desc	qty	GS11-PK	GS11-SK	location
		Tiki Torch	Bamboo tiki torch, 4 ft	2	B2	P2	W1-A9-S10-B52
P2	P2	name	desc	qty	location	reorderAt	GS13-SK
		Tiki Torch	Bamboo tiki torch, 4 ft	656	W1-A9-S10-B52	100	/GardenOutdoor/OutdoorDecor/Lighting/LanternsT
	B1	name	desc	qty	GS11-PK	GS11-SK	location
		Tiki Statue - Pele	Tiki of the Hawaiian Fire Goddess Pele, 5 ft.	1	B1	P3	W1-A15-S6-B27

## Fase 6. Crea le interrogazioni sui dati

### Obiettivo

- Crea le query principali per convalidare il modello di dati.

### Processo

- L'ingegnere del database crea manualmente una tabella DynamoDB nella AWS regione o sul proprio computer (DynamoDB Local).
- L'ingegnere del database aggiunge dati di esempio alla tabella DynamoDB.
- [L'ingegnere del database crea sfaccettature utilizzando NoSQL Workbench per Amazon DynamoDB o l'SDK per AWS Java o Python per creare query di esempio \(vedi il post del blog\).](#)

I facet sono come una vista della tabella DynamoDB.

- L'ingegnere di database e lo sviluppatore del cloud creano query di esempio utilizzando AWS Command Line Interface (AWS CLI) o AWS SDK per la lingua preferita.

## Strumenti e risorse

- Un AWS account attivo, per accedere alla console DynamoDB
- [DynamoDB Local](#) (opzionale), se desideri creare il database sul tuo computer senza accedere al servizio web DynamoDB
- [NoSQL Workbench per Amazon DynamoDB \(download e documentazione\)](#)
- [AWS SDK](#) nel linguaggio che preferisci (PythonJavaScript, PHP, .NET, Ruby, Java, Go, Node.js, C++ e SAP ABAP)

## TRACI

Utente aziendale	Business analyst	Architetto di soluzioni	Ingegnere di database	Sviluppatore di applicazioni	DevOps ingegnere
I	I	I	R/A	R	

## Output

- Codice per interrogare la tabella DynamoDB

## Esempi

- [Esempi di DynamoDB che utilizzano AWS I' SDK for Java](#)
- [Esempi in Python](#)
- [JavaScriptesempi](#)

## Fase 7. Convalida il modello di dati

### Obiettivo

- Assicurati che il modello di dati soddisfi i tuoi requisiti.

## Processo

- L'ingegnere del database popola la tabella DynamoDB con dati di esempio.
- L'ingegnere del database esegue il codice per interrogare la tabella DynamoDB.
- L'ingegnere del database raccoglie i risultati delle query.
- L'ingegnere del database raccoglie le metriche delle prestazioni delle query.
- L'utente aziendale verifica che i risultati delle query soddisfino le esigenze aziendali.
- Gli analisti aziendali convalidano i requisiti tecnici.

## Strumenti e risorse

- Un AWS account attivo, per accedere alla console DynamoDB
- [DynamoDB Local](#) (opzionale), se desideri creare il database sul tuo computer senza accedere al servizio web DynamoDB
- [AWSSDK](#) nella lingua che preferisci

## TRACI

Utente aziendale	Business analyst	Architetto di soluzioni	Ingegnere di database	Sviluppatore di applicazioni	DevOps ingegnere
A	R	I	C		

## Output

- Modello di dati approvato

## Fase 8. Rivedi la stima dei costi

### Obiettivi

- [Definisci il modello di capacità e stima i costi di DynamoDB per affinare la stima dei costi dalla fase 2.](#)
- Ottieni l'approvazione finanziaria finale dall'analista aziendale e dagli stakeholder.

### Processo

- L'ingegnere del database identifica la stima del volume di dati.
- L'ingegnere del database identifica i requisiti di trasferimento dei dati.
- L'ingegnere del database definisce le unità di capacità di lettura e scrittura richieste.
- L'analista aziendale decide tra modelli di capacità [su richiesta e modelli di capacità forniti.](#)
- L'ingegnere del database identifica la necessità della scalabilità automatica di [DynamoDB.](#)
- L'ingegnere del database inserisce i parametri nello strumento Simple Monthly Calculator.
- L'ingegnere del database presenta la stima finale del prezzo agli stakeholder aziendali.
- Gli analisti aziendali e le parti interessate approvano o rifiutano la soluzione.

### Strumenti e risorse

- [AWS Pricing Calculator](#)

### TRACI

Utente aziendale	Business analyst	Architetto di soluzioni	Ingegnere di database	Sviluppatore di applicazioni	DevOps ingegnere
C	A	I	R		

## Output

- modello di capacità
- Stima dei costi rivista

## Fase 9. Implementa il modello di dati

### Obiettivo

- Distribuisci la tabella (o le tabelle) di DynamoDB su. Regione AWS

### Processo

- DevOps architect crea un AWS CloudFormation modello o un altro strumento Infrastructure as Code (IaC) per la tabella (o le tabelle) DynamoDB. AWS CloudFormation fornisce un modo automatizzato per effettuare il provisioning e configurare le tabelle e le risorse associate.

### Strumenti e risorse

- [AWS CloudFormation](#)

### TRACI

Utente aziendale	Business analyst	Architetto di soluzioni	Ingegnere di database	Sviluppatore di applicazioni	DevOps ingegnere
I	I	C	C		R/A

### Output

- Modello di AWS CloudFormation

## Esempio

```
mySecondDDBTable:
  Type: AWS::DynamoDB::
  Table DependsOn: "myFirstDDBTable"
  Properties:
    AttributeDefinitions:
      - AttributeName: "ArtistId"
        AttributeType: "S"
      - AttributeName: "Concert"
        AttributeType: "S"
      - AttributeName: "TicketSales"
        AttributeType: "S"
    KeySchema:
      - AttributeName: "ArtistId"
        KeyType: "HASH"
      - AttributeName: "Concert"
        KeyType: "RANGE"
    ProvisionedThroughput:
      ReadCapacityUnits:
        Ref: "ReadCapacityUnits"
      WriteCapacityUnits:
        Ref: "WriteCapacityUnits"
    GlobalSecondaryIndexes:
      - IndexName: "myGSI"
        KeySchema:
          - AttributeName: "TicketSales"
            KeyType: "HASH"
        Projection:
          ProjectionType: "KEYS_ONLY"
        ProvisionedThroughput:
          ReadCapacityUnits:
            Ref: "ReadCapacityUnits"
          WriteCapacityUnits:
            Ref: "WriteCapacityUnits"
    Tags:
      - Key: mykey
        Value: myvalue
```



# Modelli

I modelli forniti in questa sezione si basano sulla [modellazione dei dati dei giocatori di gioco con Amazon DynamoDB AWS sul sito Web](#).

## Note

Le tabelle di questa sezione utilizzano MM come abbreviazione per milione e K come abbreviazione per mille.

## Argomenti

- [Modello di valutazione dei requisiti aziendali](#)
- [Modello di valutazione dei requisiti tecnici](#)
- [Modello di modelli di accesso](#)

## Modello di valutazione dei requisiti aziendali

Fornisci una descrizione del caso d'uso:

## Descrizione

Immagina di creare un gioco multigiocatore online. Nella tua partita, gruppi di 50 giocatori si uniscono a una sessione per giocare una partita, che in genere richiede circa 30 minuti. Durante la partita, devi aggiornare il record di un giocatore specifico per indicare il periodo di tempo in cui il giocatore ha giocato, le sue statistiche o se ha vinto la partita. Gli utenti vogliono vedere le partite precedenti a cui hanno giocato, per vedere i vincitori delle partite o per guardare un replay delle azioni di ogni partita.

Fornisci informazioni sui tuoi utenti:

Utente	Descrizione	Numero previsto
Giocatore	Giocatore di giochi online.	1 MM

Team di sviluppo	Team interno che utilizzerà le statistiche di gioco per migliorare il  esperienza di gioco.	100
------------------	---	-----

Fornisci informazioni sulle fonti dei dati e su come i dati verranno acquisiti:

Origine	Descrizione	Utente
Gioco online	I giocatori creeranno profili e inizieranno nuove partite.	Giocatore
App di gioco	L'app di gioco raccoglierà automaticamente statistiche sui giochi, come l'ora di inizio e di fine, il numero di giocatori, la posizione di ogni giocatore e la mappa del gioco.	

Fornisci informazioni su come verranno consumati i dati:

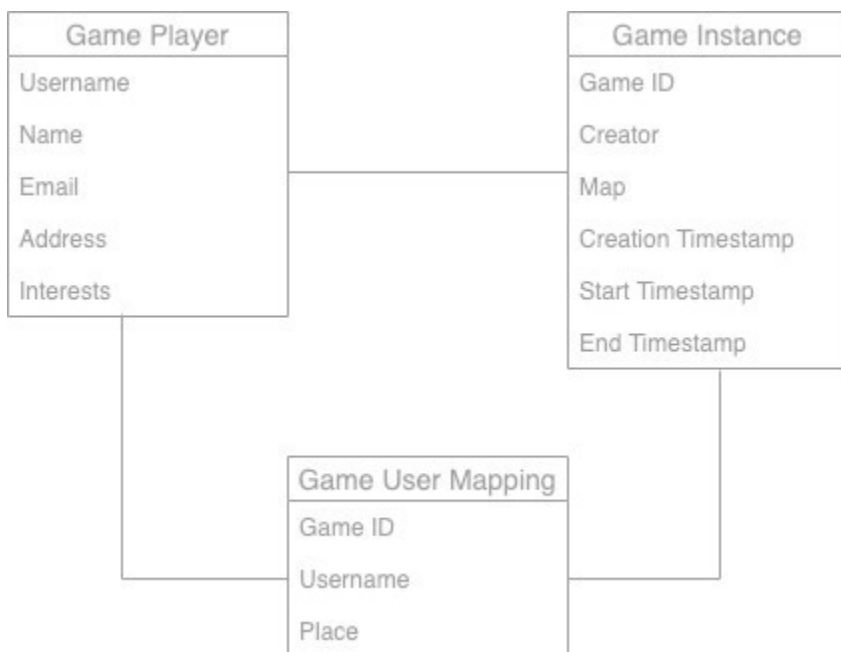
Consumatore	Descrizione	Utente
Gioco online	I giocatori vedranno i profili e rivedranno le loro statistiche di gioco.	Giocatore
Analisi dei dati	Il team di sviluppo del gioco estrarrà le statistiche di gioco per l'analisi dei dati e per migliorare l'esperienza dell'utente. I dati verranno esportati dal data store e importati in Amazon S3 per	Team di sviluppo

supportare l'analisi tramite un'applicazione Spark.

Fornisci un elenco di entità e il modo in cui vengono identificate:

Nome dell'entità	Descrizione	Identifier (Identificatore)
Giocatore di gioco	Memorizza informazioni come identificazione, indirizzo, dati demografici, interessi per ogni utente (giocatore).	Nome utente
Istanza di gioco	Fornisce informazioni su ogni partita giocata, tra cui creatore, inizio, fine e mappa Yplayed.	ID del gioco
Mappatura degli utenti del gioco	Rappresenta le many-to-many relazioni tra utenti e giochi.	ID e nome utente del gioco

Crea un modello ER per le entità:



Fornisci statistiche di alto livello sulle entità:

Nome entità	Numero stimato di record	Dimensione del record	Note
Giocatore di gioco	1 MM	< 1 KB	La piattaforma di gioco ha circa 1 milione di utenti.
Istanza di gioco	6 MM (100.000 K/giorno* 60 giorni)	< 1 KB	In media, ci sono 100.000 partite ogni giorno. Dobbiamo memorizzare gli ultimi 60 giorni.
Mappatura degli utenti del gioco	300 MM (giochi da 6 MM* 50 giocatori)	< 1 KB	In media, ogni partita ha 50 giocatori sui quali dobbiamo memorizzare informazioni.

## Modello di valutazione dei requisiti tecnici

Fornisci informazioni sui tipi di inserimento dei dati:

Tipo di ingestione dei dati	Y/N	Descrizione	Frequency (Frequenza)
Accesso all'applicazione	Y		
Gateway API	Y		
Streaming di dati	N		
Processo Batch	N		
ETL	N		

Importazione dei dati	N
Serie temporali	N

Fornisci informazioni sui tipi di consumo di dati:

Tipo di consumo di dati	Y/N	Descrizione	Frequency (Frequenza)
Accesso all'applicazione			
Gateway API			
Esportazione dei dati			
Analisi dei dati			
Aggregazione dei dati			
Creazione di report			
Cerca			
Streaming di dati			
ETL			

Fornisci stime del volume di dati:

Nome dell'entità	Numero stimato di record	Dimensione del record	Volume di dati
Giocatore di gioco	1 MM	< 1 KB	~ 1 GB (1 MM X 1 KB)
Istanza di gioco	6 MM	< 1 KB	~ 6 GB

	(100K/giorno* 60 giorni)		(6 MM X 1 KB)
Mappatura degli utenti del gioco	300 MM  (giochi da 6 MM* 50 giocatori)	< 1 KB	~ 300 GB  (300 MM* 1 KB)

### Note

Il periodo di conservazione dei dati è di 60 giorni. Dopo 60 giorni, i dati devono essere archiviati in Amazon S3 per l'analisi, utilizzando [DynamoDB Time to Live \(TTL\)](#) per spostare automaticamente i dati da DynamoDB ad Amazon S3.

Rispondi a queste domande sugli schemi temporali:

- In che periodo l'applicazione è disponibile per l'utente (ad esempio, 24 ore su 24, 7 giorni su 7 o dalle 9:00 alle 17:00 nei giorni feriali)?
- C'è un picco di utilizzo durante il giorno? Quante ore? Qual è la percentuale di utilizzo dell'applicazione?

Specificate i requisiti di velocità di scrittura:

Nome dell'entità	Scritture al giorno	Ore/giorno	Scritture/secondo
Giocatore di gioco	10.000 aggiornamenti	18	< 1
Istanza di gioco	300.000	18	< 5
Mappatura degli utenti del gioco	1.800.000.000	18	~ 27,77

### Note

Operazioni di scrittura di Game Player: l'1% degli utenti aggiorna il proprio profilo ogni giorno, quindi prevediamo 10.000 aggiornamenti per 1.000.000 di utenti.

Operazioni di scrittura su istanze di gioco: 100.000 giochi/giorno. Per ogni gioco abbiamo almeno 3 operazioni di scrittura, alla creazione, all'inizio e alla fine, per un totale di 300.000 operazioni di scrittura.

Operazioni di scrittura di Game User Mapping: 100.000 partite al giorno per ogni partita con 50 giocatori. La durata media del gioco è di 30 minuti e la posizione del giocatore viene aggiornata ogni 5 secondi. Stimiamo una media di 360 aggiornamenti per giocatore, quindi il totale è  $100.000 * 50 * 360 = 1.800.000.000$  di operazioni di scrittura.

Specificate i requisiti di velocità di lettura:

Nome dell'entità	Lecture/giorno	Ore/giorno	Lecture/sec
Giocatore di gioco	200.000	18	~ 3
Istanza di gioco	5.000.000	18	~ 77
Mappatura degli utenti del gioco	1.800.000.000	18	~ 27,77

#### Note

Operazioni di lettura da parte di Game Player: il 20 per cento degli utenti inizia a giocare, quindi  $1 \text{ MM} * 0,2 = 200.000$ .

Operazioni di lettura delle istanze di gioco: 100.000 giochi/giorno. Per ogni gioco abbiamo almeno 1 operazione di lettura per giocatore e 50 giocatori per partita, quindi il totale è di 5.000.000 di operazioni di lettura.

Operazioni di lettura di Game User Mapping: 100.000 partite al giorno per 50 giocatori. La durata media del gioco è di 30 minuti e la posizione del giocatore viene aggiornata ogni 5 secondi. Stimiamo una media di 360 aggiornamenti per giocatore e ogni aggiornamento richiede un'operazione di lettura, quindi il totale è  $100.000 * 50 * 360 = 1.800.000.000$  di operazioni di lettura.

Specificare i requisiti di latenza di accesso ai dati:

Operazioni	99 percentili	Latenza massima
Lettura	30 ms	100 ms
Scrittura	10 ms	50 ms

Specificare i requisiti di disponibilità dei dati:

Requisito	Y/N	Parametro	Note
Alta disponibilità	Y	99,9%	
RTO	Y	1 ora	Obiettivo del tempo di recupero
RPO	Y	1 ora	Obiettivo del punto di ripristino
Ripristino di emergenza	N		
Replica dei dati a livello regionale	N		
Replica dei dati tra regioni	N	Latenza di 3 secondi	QualiRegioni AWS?

Specificare i requisiti di sicurezza:

Requisito	Y/N	Note
Archivio dati sensibili	N	Informazioni sanitarie protette (PHI), informazioni sul settore delle carte di pagamento (PCI), informazioni di identificazione personale (PII)?



Crittografia dei dati inattivi	Y
Crittografia in transito	Y
Crittografia lato client	N
Qualsiasi libreria di crittografia proprietaria o di terze parti	N
Registrazione dell'accesso ai dati	N
Controllo dell'accesso ai dati	N

## Modello di modelli di accesso

Raccogli e documenta le informazioni sui modelli di accesso per il caso d'uso utilizzando i seguenti campi:

Campo	Descrizione
Schema di accesso	Fornire un nome per il modello di accesso.
Descrizione	Fornire una descrizione più dettagliata del modello di accesso.
Priority (Priorità)	Definire una priorità per il modello di accesso (alta, media o bassa). Questo definisce i modelli di accesso più rilevanti per l'applicazione.
Leggi o scrivi	È un modello di accesso in lettura o scrittura?
Type	Il pattern accede a un singolo elemento, a più elementi o a tutti gli elementi?
Filtro	Il modello di accesso richiede un filtro?
Ordina	Il risultato richiede un ordinamento?

## Modello

Schema di accesso	Descrizione	Priority (Priorità)	Leggi o scrivi	Tipo (articolo singolo), multiplo articoli o tutti)	Attributo chiave	Filtri	Ordinamento dei risultati
Crea un profilo utente	L'utente crea un nuovo profilo.	Elevate	Scrittura	Elemento singolo	Nome utente	N/D	N/D
Aggiorna il profilo utente	L'utente aggiorna il proprio profilo.	Medio	Scrittura	Elemento singolo	Nome utente	Nome utente = utente corrente	N/D
Ottieni il profilo utente	L'utente esamina il proprio profilo.	Elevate	Lettura	Elemento singolo	Nome utente	Nome utente = utente corrente	N/D
Crea un gioco	L'utente crea un nuovo gioco.	Elevate	Scrittura	Elemento singolo	ID del gioco	N/D	N/D
Trova giochi aperti	L'utente cerca giochi aperti. I risultati della ricerca	Elevate	Lettura	Elementi multipli		GameStatus = aperto	Timestamp di inizio discendente

vengono  
ordinati  
per data  
e ora di  
inizio in  
ordine  
decrescen  
te.

Trova giochi aperti tramite mappa	L'utente cerca i giochi aperti utilizzando una mappa specifica ordinata in ordine decrescente per data e ora di inizio ordine.	Medio	Lettura	Articoli multipli	GameStatus = aprire Mappa = XYZ	Timestamp di inizio discendente	
Visualizza il gioco	L'utente esamina i dettagli di un gioco.	Elevate	Lettura	Elemento singolo	ID del gioco	N/D	N/D

Visualizza gli utenti in un gioco	L'utente ottiene un elenco di tutti gli utenti di un gioco.	Medio	Lettura	Oggetti multipli		GameID = XYZ	N/D
Unisci un utente a una partita	L'utente si unisce a una partita aperta.	Elevate	Scrittura	Elemento singolo	GameID e nome utente	GameStatus = aperto	N/D
Inizia una partita	L'utente inizia una nuova partita.	Elevate	Scrittura	Elemento singolo	ID del gioco	N/D	N/D
Aggiorna il gioco per l'utente	Aggiorna la posizione dell'utente nel gioco.	Medio	Scrittura	Elemento singolo	GameID e nome utente	N/D	N/D
Aggiorna il gioco	La partita termina; aggiorna le statistiche.	Medio	Scrittura	Elemento singolo	ID del gioco	N/D	N/D

Trova tutti i giochi precedent i per un utente	Elenca tutte le partite giocate da un utente in base alla data di inizio della partita.	Bassa	Lettura	Oggetti multipli	Nome utente e GameID	Nome utente = utente corrente	Timestamp di inizio
Esporta dati per l'analisi dei dati	Il team di sviluppo eseguirà un processo in batch per esportare i dati in Amazon S3.	Bassa	Lettura	All (Tutti)	N/D	N/D	N/D

# Best practice

Prendi in considerazione l'utilizzo delle seguenti best practice di progettazione di DynamoDB:

- [Progettazione delle chiavi di partizione](#): utilizza una chiave di partizione ad alta cardinalità per distribuire il carico in modo uniforme.
- Modello di [progettazione dell'elenco di adiacenza](#): [utilizza questo modello](#) di progettazione per la gestione one-to-many e le relazioni. many-to-many
- Indice [sparso](#): [utilizza l'indice](#) sparso per i tuoi indici secondari globali (GSI). Quando crei un GSI, specifica una chiave di partizione e opzionalmente una chiave di ordinamento. Solo gli elementi della tabella di base che contengono una chiave di partizione GSI corrispondente vengono visualizzati nell'indice sparse. Questo aiuta a mantenere i GSI più piccoli.
- [Sovraccarico dell'indice](#): utilizza lo stesso GSI per indicizzare vari tipi di articoli.
- [Partizionamento in scrittura del GSI](#): partiziona con cura per distribuire i dati tra le partizioni per query efficienti e veloci.
- [Oggetti di grandi dimensioni](#): archivia solo i metadati all'interno della tabella, salva il blob in Amazon S3 e conserva il riferimento in DynamoDB. Suddividi gli elementi di grandi dimensioni in più elementi e indicizzali in modo efficiente utilizzando le chiavi di ordinamento.

Per ulteriori best practice di progettazione, consulta la documentazione di [Amazon DynamoDB](#).

# Esempio di modellazione gerarchica dei dati

Le sezioni seguenti utilizzano un esempio di azienda automobilistica per mostrare come utilizzare le fasi del processo di modellazione dei dati per progettare un sistema di gestione dei componenti a più livelli in DynamoDB.

## Argomenti

- [Fase 1: Identifica i casi d'uso e il modello logico di dati](#)
- [Fase 2: Creare una stima preliminare dei costi](#)
- [Fase 3: Identifica i tuoi modelli di accesso ai dati](#)
- [Fase 4: Identificare i requisiti tecnici](#)
- [Fase 5: Creare un modello di dati DynamoDB](#)
- [Fase 6: Creare interrogazioni di dati](#)
- [Fase 7: Convalida del modello di dati](#)
- [Fase 8: Rivedere la stima dei costi](#)
- [Fase 9: Implementazione del modello di dati](#)

## Fase 1: Identifica i casi d'uso e il modello logico di dati

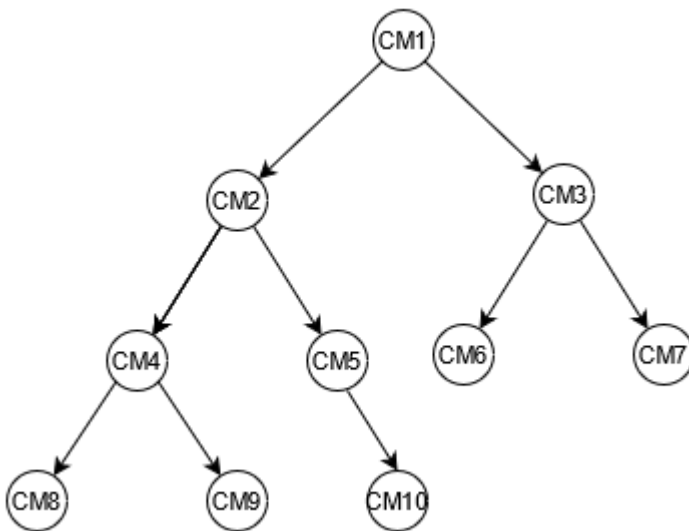
Un'azienda automobilistica desidera creare un sistema di gestione transazionale dei componenti per archiviare e cercare tutti i ricambi auto disponibili e creare relazioni tra diversi componenti e parti. Ad esempio, un'automobile contiene più batterie, ogni batteria contiene più moduli di alto livello, ogni modulo contiene più celle e ogni cella contiene più componenti di basso livello.

In genere, per la creazione di un modello di relazione gerarchico, la scelta migliore è quella di utilizzare un database a grafo come [Amazon Neptune](#). In alcuni casi, tuttavia, Amazon DynamoDB è un'alternativa migliore per la modellazione gerarchica dei dati grazie alla sua flessibilità, sicurezza, prestazioni e scalabilità.

Ad esempio, potresti creare un sistema in cui l'80-90% delle query siano transazionali, in cui DynamoDB si adatta bene. In questo esempio, l'altro 10-20 percento delle query sono relazionali, dove un database grafico come Neptune si adatta meglio. In questo caso, l'inclusione di un database aggiuntivo nell'architettura per soddisfare solo il 10-20 percento delle query potrebbe

umentare i costi. Inoltre, aggiunge l'onere operativo legato alla manutenzione di più sistemi e alla sincronizzazione dei dati. È invece possibile modellare le query relazionali del 10-20% in DynamoDB.

La creazione di un diagramma di un albero di esempio per i componenti dell'automobile può aiutarti a mappare la relazione tra di essi. Il diagramma seguente mostra un grafico delle dipendenze con quattro livelli. CM1 è il componente di livello superiore dell'automobile di esempio stessa. Ha due sottocomponenti per due batterie di esempio, CM2 e CM3. Ogni batteria ha due sottocomponenti, che sono i moduli. CM2 ha i moduli CM4 e CM5 e CM3 ha i moduli CM6 e CM7. Ogni modulo ha diversi sottocomponenti, che sono le celle. Il modulo CM4 ha due celle, CM8 e CM9. CM5 ha una cella, CM10. CM6 e CM7 non hanno ancora celle associate.



Questa guida utilizzerà questo albero e gli identificatori dei suoi componenti come riferimento. Un componente principale verrà denominato padre e un sottocomponente verrà denominato figlio. Ad esempio, il componente principale CM1 è il padre di CM2 e CM3. CM2 è il padre di CM4 e CM5. Questo grafico rappresenta le relazioni padre-figlio.

Dall'albero, puoi vedere il grafico completo delle dipendenze di un componente. Ad esempio, CM8 dipende da CM4, che dipende da CM2, che dipende da CM1. L'albero definisce il grafico completo delle dipendenze come percorso. Un percorso descrive due cose:

- Il grafico delle dipendenze
- La posizione nell'albero

Compilazione dei modelli per i requisiti aziendali:



Fornisci informazioni sui tuoi utenti:

Utente	Descrizione
Dipendente	Dipendente interno dell'azienda automobilistica che necessita di informazioni sulle auto e sui relativi componenti

Fornisci informazioni sulle fonti dei dati e su come i dati verranno acquisiti:

Origine	Descrizione	Utente
Sistema di gestione	Sistema che memorizzerà tutti i dati relativi ai ricambi auto disponibili e alle loro relazioni con altri componenti e parti.	Impiegato

Fornisci informazioni su come verranno consumati i dati:

Consumatore	Descrizione	Utente
Sistema di gestione	Recupera tutti i componenti secondari immediati per un ID del componente principale.	Dipendente
Sistema di gestione	Recupera un elenco ricorsivo di tutti i componenti secondari per l'ID di un componente.	Impiegato
Sistema di gestione	Visualizza gli antenati di un componente.	Impiegato

## Fase 2: Creare una stima preliminare dei costi

È importante calcolare una stima del costo per tutti gli ambienti dell'applicazione in modo da poter verificare se la soluzione è economicamente sostenibile. Una buona pratica consiste nell'effettuare

una stima di alto livello e ottenere l'approvazione dell'analista aziendale prima di procedere con lo sviluppo e l'implementazione.

- L'ingegnere del database crea l'analisi iniziale dei costi utilizzando le informazioni disponibili e gli esempi presentati nella pagina dei prezzi di [DynamoDB](#).
  - [Crea una stima dei costi per la capacità su richiesta \(vedi esempio\)](#).
  - Crea una stima dei costi per la capacità assegnata (vedi [esempio](#)).
    - Per il modello di capacità assegnata, ottieni il costo stimato dal calcolatore e applica lo sconto per la capacità riservata.
  - Confrontate i costi stimati dei due modelli di capacità.
  - Crea una stima per tutti gli ambienti (Dev, Prod, QA).
- L'analista aziendale esamina e approva o rifiuta la stima preliminare dei costi.

Utilizzando questi valori di riferimento, è possibile creare un prezzo stimato da inviare per l'approvazione. Per creare il budget, puoi utilizzare la [pagina dei prezzi di DynamoDB e AWS Pricing Calculator](#).

## Fase 3: Identifica i tuoi modelli di accesso ai dati

Questo caso d'uso di esempio presenta i seguenti modelli di accesso per la gestione delle relazioni tra diversi componenti dell'auto.

Schema di accesso	Priority (Priorità)	Leggi o scrivi	Descrizione	Type	Filtri	Ordinamento dei risultati
Bambino immediato	Elevata	Lettura	Recupera tutti i componenti secondari immediati per un ID del component	Più di uno	Component ID	N/D

			e principal e.			
Tutti i component i secondari	Elevata	Lettura	Recupera un elenco ricorsivo di tutti i component i secondari per l'ID di un component e.	Più di uno	Component ID	N/D
Antenati	Elevata	Lettura	Recupera gli antenati di un component e.	Più di uno	Component ID	N/D

## Fase 4: Identificare i requisiti tecnici

Questo esempio non ha requisiti tecnici specifici, che non rientrano nell'ambito di questo esempio. In casi reali, è consigliabile completare questo passaggio e verificare che tutti i requisiti tecnici siano soddisfatti prima di procedere con lo sviluppo e l'implementazione. Puoi utilizzare il [questionario di esempio](#) per completare questo passaggio nel tuo business case. Inoltre, consigliamo di convalidare le [quote dei servizi DynamoDB](#) per assicurarsi che non vi siano limiti rigidi nella soluzione progettata.

## Fase 5: Creare un modello di dati DynamoDB

Definite le chiavi di partizione per la tabella di base e gli indici secondari globali (GSI):

- Seguendo le migliori pratiche di progettazione chiave, utilizzate ComponentId come chiave di partizione per la tabella di base in questo esempio. Poiché è unica, ComponentId può offrire granularità. DynamoDB utilizza il valore hash della chiave di partizione per determinare la partizione in cui i dati vengono archiviati fisicamente. L'ID univoco del componente genera un

valore hash diverso, che può facilitare la distribuzione dei dati all'interno della tabella. È possibile interrogare la tabella di base utilizzando una chiave di ComponentId partizione.

- Per trovare i figli immediati di un componente, crea un GSI in cui si ParentId trova la chiave di partizione e ComponentId la chiave di ordinamento. È possibile interrogare questo GSI utilizzando ParentId come chiave di partizione.
- Per trovare tutti i figli ricorrenti di un componente, crea un GSI dove GraphId è la chiave di partizione, e Path è la chiave di ordinamento. Puoi eseguire query su questo GSI utilizzando GraphId come chiave di partizione e l'operatore BEGINS\_WITH(Path, "\$path") sulla chiave di ordinamento.

	Chiave di partizione	Chiave di ordinamento	Attributi di mappatura
Tabella di base	ComponentId		ParentId, GraphId, Path
GSI1	ParentId	ComponentId	
GSI2	GraphId	Path	ComponentId

## Archiviazione dei componenti nella tabella

Il passaggio successivo consiste nell'archiviare ogni componente nella tabella di base di DynamoDB. Dopo aver inserito tutti i componenti dall'albero di esempio, si ottiene la seguente tabella di base.

ComponentId	ParentId	GraphId	Path
CM1		CM1#1	CM1
CM2	CM1	CM1#1	CM1 CM2

CM3	CM1	CM1#1	CM1 CM3
CM4	CM2	CM1#1	CM1 CM2 CM4
CM5	CM2	CM1#1	CM1 CM2 CM5
CM6	CM3	CM1#1	CM1 CM3 CM6
CM7	CM3	CM1#1	CM1 CM3 CM7
CM8	CM4	CM1#1	CM1 CM2 CM4 CM8
CM9	CM4	CM1#1	CM1 CM2 CM4 CM9
CM10	CM5	CM1#1	CM1 CM2 CM5 CM10

## L'indice GSI1

Per controllare tutti i figli immediati di un componente, create un indice da utilizzare `ParentId` come chiave di partizione e `ComponentId` come chiave di ordinamento. La seguente tabella pivot rappresenta l'indice GSI1. Puoi utilizzare questo indice per recuperare tutti i componenti figlio immediati utilizzando un ID del componente principale. Ad esempio, puoi scoprire quante batterie sono disponibili in un'automobile (CM1) o quali celle sono disponibili in un modulo (CM4).

ParentId	ComponentId
CM1	CM2

	CM3
CM2	CM4
	CM5
CM3	CM6
	CM7
CM4	CM8
	CM9
CM5	CM10

## L'indice GSI2

La seguente tabella pivot rappresenta l'indice GSI2. È configurato utilizzando GraphId come chiave di partizione e Path come chiave di ordinamento. Usando GraphId e l'begin\_withoperazione sulla chiave di ordinamento (Path), puoi trovare la discendenza completa di un componente in un albero.

GraphId	Path	ComponentId
CM1#1	CM1	CM1
	CM1 CM2	CM2
	CM1 CM3	CM3
	CM1 CM2 CM4	CM4
	CM1 CM2 CM5	CM5
	CM1 CM2 CM4 CM8	CM8
	CM1 CM2 CM4 CM9	CM9
	CM1 CM2 CM5 CM10	CM10

CM1 CM3 CM6	CM6
CM1 CM3 CM7	CM7

## Fase 6: Creare interrogazioni di dati

Dopo aver definito i modelli di accesso e progettato il modello di dati, è possibile interrogare i dati gerarchici nel database DynamoDB. Come best practice per risparmiare sui costi e contribuire a garantire le prestazioni, negli esempi seguenti viene utilizzata solo l'operazione di interrogazione che non è prevista. Scan

- Trova gli antenati di un componente.

Per trovare gli antenati (padre, nonno, bisnonno e così via) del componente CM8, esegui una query sulla tabella di base usando `ComponentId = "CM8"`. La query restituirà il seguente record.

Per ridurre la dimensione dei dati dei risultati, puoi utilizzare un'espressione di proiezione per restituire solo l'attributo `Path`.

ComponentId	ParentId	GraphId	Path
CM8	CM4	CM1#1	CM1 CM2 CM4 CM8

Path

CM1|CM2|CM4|CM8

Ora, dividi il percorso usando la pipe («|») e prendi i primi componenti N-1 per ottenere gli antenati.

Risultato della query: gli antenati di CM8 sono CM1, CM2, CM4.

- Trova i figli immediati di un componente.

Per ottenere tutti i componenti secondari immediati, o quelli a valle di un livello, per il componente CM2, interroga GSI1 utilizzando `ParentId = "CM2"` La query restituirà il seguente record.

ParentId	ComponentId
CM2	CM4
	CM5

- Trova tutti i componenti secondari a valle utilizzando un componente di primo livello.

Per ottenere tutti i componenti secondari o a valle del componente di primo livello CM1, interrogate GSI2 utilizzando `GraphId = "CM1#1"` and `e e begins_with("Path", "CM1|")` utilizzate un'espressione di proiezione con. `ComponentId` Restituirà tutti i componenti relativi a quell'albero.

Questo esempio ha un solo albero in cui il componente di primo livello è CM1. In realtà, potresti avere milioni di componenti di livello superiore nella stessa tabella.

GraphId	ComponentId
	CM2
CM1#1	CM3
	CM4
	CM5
	CM8
	CM9
	CM10
	CM6
	CM7

- Trova tutti i componenti secondari a valle utilizzando un componente di livello intermedio.

Per ottenere tutti i componenti secondari, o a valle, in modo ricorsivo per il componente CM2, hai due opzioni. Puoi eseguire query ricorrenti livello per livello oppure eseguire query sull'indice GSI2.



- Esegui una query su GSI1, livello per livello, in modo ricorrente, fino a raggiungere l'ultimo livello di componenti figlio.

1. Esegui una query su GSI1 tramite `ParentId = "CM2"`. Restituirà il seguente record.

ParentId	ComponentId
CM2	CM4
	CM5

2. Esegui di nuovo una query su GSI1 tramite `ParentId = "CM4"`. Restituirà il seguente record.

ParentId	ComponentId
CM4	CM8
	CM9

3. Esegui di nuovo una query su GSI1 tramite `ParentId = "CM5"`. Restituirà il seguente record.

Continua il ciclo: esegui una query per ciascun `ComponentId` fino a raggiungere l'ultimo livello. Quando una query che utilizza `ParentId = "<ComponentId>"` non restituisce alcun risultato, il risultato precedente proveniva dall'ultimo livello dell'albero.

ParentId	ComponentId
CM5	CM10

4. Unisci tutti i risultati.

```
risultato= [CM4, CM5] + [CM8, CM9] + [CM10]
          = [CM4, CM5, CM8, CM9, CM10]
```

- Esegui una query su GSI2, che archivia un albero gerarchico per un componente di livello superiore (un'automobile o CM1).

1. Innanzitutto, trova il componente di livello superiore o l'antenato principale e Path di CM2. Per farlo, esegui una query sulla tabella di base usando `ComponentId = "CM2"` per trovare il percorso di quel componente nell'albero gerarchico. Seleziona gli attributi e `h. GraphId Pat`. La query restituirà il seguente record.

GraphId	Path
CM1#1	CM1 CM2

2. Interroga GSI2 utilizzando `GraphId = "CM1#1" AND BEGINS_WITH("Path", "CM1|CM2|")`. La query restituirà i seguenti risultati.

GraphId	Path	ComponentId
CM1#1	CM1 CM2 CM4	CM4
	CM1 CM2 CM5	CM5
	CM1 CM2 CM4 CM8	CM8
	CM1 CM2 CM4 CM9	CM9
	CM1 CM2 CM5 CM10	CM10

3. Seleziona l'attributo `ComponentId` per restituire tutti i componenti figlio per CM2.

## Fase 7: Convalida del modello di dati

In questo passaggio, l'utente aziendale convalida i risultati della query e verifica se soddisfano le esigenze aziendali. È possibile utilizzare la tabella seguente per verificare i modelli di accesso rispetto ai requisiti dell'utente.

Domanda	Tabella base/GSI	Query
Come utente, voglio recuperare tutti i componenti figlio immediati per un ID componente padre.	GSI1	<code>ParentId = "&lt;ComponentId&gt;"</code>

(Trova i figli immediati di un componente).

```
GSI1: ParentId =
"<ComponentId>"
```

oppure

```
GSI2: GraphId =
"<TopLevelComponentId>#N" AND BEGINS_WITH("Path", "<PATH_OF_Component>")
```

(Trova tutti i componenti figlio di livello inferiore utilizzando un componente e di livello superiore. Trova tutti i componenti figlio di livello inferiore utilizzando un componente di livello intermedio).

```
ComponentId =
"<ComponentId>" , quindi
seleziona l'attributo Path.
```

(Trova gli antenati di un componente).

Come utente, voglio recuperare e un elenco ricorsivo di tutti i componenti figlio per un ID componente.

GSI1 o GSI2

Come utente, voglio vedere gli antenati di un componente.

Tabella di base

È inoltre possibile implementare uno script (test) in qualsiasi linguaggio di programmazione per interrogare direttamente DynamoDB e confrontare i risultati con quelli previsti.

## Fase 8: Rivedere la stima dei costi

Rivedi e perfeziona nuovamente la stima dei costi. Inoltre, è buona norma convalidarla con gli stakeholder aziendali e ottenere l'approvazione per passare alla fase successiva.

## Obiettivi

- [Definisci il modello di capacità e stima i costi di DynamoDB per affinare la stima dei costi dalla fase 2.](#)
- Ottieni l'approvazione finanziaria finale dall'analista aziendale e dagli stakeholder.

## Processo

- L'ingegnere del database identifica la stima del volume di dati.
- L'ingegnere del database identifica i requisiti di trasferimento dei dati.
- L'ingegnere del database definisce le unità di capacità di lettura e scrittura richieste.
- L'analista aziendale decide tra modelli di capacità [su richiesta e modelli di capacità forniti.](#)
- L'ingegnere del database identifica la necessità della scalabilità automatica di [DynamoDB.](#)
- L'ingegnere del database inserisce i parametri in AWS Pricing Calculator
- L'ingegnere del database presenta la stima del prezzo finale agli stakeholder aziendali.
- Gli analisti aziendali e le parti interessate approvano o rifiutano la soluzione.

## Fase 9: Implementazione del modello di dati

Per questo esempio specifico, l'implementazione del modello è stata effettuata utilizzando [NoSQL Workbench](#), un'applicazione per lo sviluppo e il funzionamento di database moderni. Utilizzando questo strumento, hai la possibilità di creare un modello di dati, caricare dati e distribuirlo direttamente sul tuo Account AWS. Se si desidera implementare questo esempio, è possibile utilizzare il seguente AWS CloudFormation modello, generato da NoSQL Workbench.

```
AWSTemplateFormatVersion: 2010-09-09
Resources:
  Components:
    Type: 'AWS::DynamoDB::Table'
    Properties:
      KeySchema:
        - AttributeName: ComponentId
          KeyType: HASH
      AttributeDefinitions:
        - AttributeName: ComponentId
          AttributeType: S
```

```
- AttributeName: ParentId
  AttributeType: S
- AttributeName: GraphId
  AttributeType: S
- AttributeName: Path
  AttributeType: S
GlobalSecondaryIndexes:
- IndexName: GS1
  KeySchema:
  - AttributeName: ParentId
    KeyType: HASH
  - AttributeName: ComponentId
    KeyType: RANGE
  Projection:
    ProjectionType: KEYS_ONLY
- IndexName: GSI2
  KeySchema:
  - AttributeName: GraphId
    KeyType: HASH
  - AttributeName: Path
    KeyType: RANGE
  Projection:
    ProjectionType: INCLUDE
    NonKeyAttributes:
    - ComponentId
BillingMode: PAY_PER_REQUEST
TableName: Components
```

# Risorse aggiuntive

## Ulteriori informazioni su DynamoDB

- [Prezzi DynamoDB](#)
- [Documentazione DynamoDB](#)
- [Progettazione NoSQL per DynamoDB](#)
- [Scrivere lo sharding](#)
- [Indici secondari locali \(LSI\)](#)
- [Indici secondari globali \(GSI\)](#)
- [Sovraccarico dei GSI](#)
- [Sharding GSI](#)
- [Utilizzo dei GSI per creare una replica alla fine coerente](#)
- [Indici sparsi](#)
- [Interrogazioni di aggregazione materializzate](#)
- [Modello di progettazione delle serie temporali](#)
- [Modello di progettazione dell'elenco di adiacenze](#)
- [Modelli di capacità su richiesta e forniti](#)
- [Scalabilità automatica DynamoDB](#)
- [DynamoDB Time to Live \(TTL\)](#)
- [Modellazione dei dati dei giocatori con DynamoDB \(lab\)](#)

## Servizi AWS

- [AWS CloudFormation](#)
- [Amazon S3](#)

## Strumenti

- [AWS Pricing Calculator](#)
- [Workbench NoSQL per DynamoDB](#)
- [DynamoDB Local](#)

- [DynamoDB e AWS SDK](#)

### Best practice

- [Le migliori pratiche per la progettazione e l'architettura con DynamoDB \(documentazione DynamoDB\)](#)
- [Le migliori pratiche per l'utilizzo degli indici secondari](#) (documentazione DynamoDB)
- [Le migliori pratiche per l'archiviazione di elementi e attributi di grandi dimensioni](#) (documentazione DynamoDB)
- [Scelta della chiave di partizione DynamoDB corretta \(blog Database\) AWS](#)
- [Come progettare indici secondari Amazon DynamoDBGlobal \(blog Database\) AWS](#)
- [Cosa sono le sfaccettature in NoSQL Workbench per Amazon DynamoDB](#) (sito Web Medium)

### AWS risorse generali

- [AWS sito web Prescriptive Guidance](#)
- [Documentazione di AWS](#)
- [AWS riferimento generale](#)

# Collaboratori

I collaboratori di questa guida includono:

- Camilo Gonzalez, Senior Data Architect, AWS
- Moinul Al-Mamun, architetto senior per i Big Data, AWS
- Santiago Segura, consulente per i servizi professionali, AWS
- Satheish Kumar Chandraprakasam, architetto di applicazioni cloud, AWS



## 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="#">È stata aggiunta una sezione sulle migliori pratiche e un esempio per la modellazione gerarchica dei dati.</a>	<a href="#">Abbiamo aggiunto un riepilogo delle best practice di DynamoDB e step-by-step un esempio di progettazione e convalida di un modello gerarchico.</a>	5 dicembre 2023
<a href="#">Pubblicazione iniziale</a>	—	26 ottobre 2020

# AWS Glossario delle linee guida prescrittive

I seguenti sono termini comunemente usati 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:

- **Rifattorizzare/riprogettare:** trasferisci un'applicazione e modifica la sua architettura 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: esegui la migrazione del database Oracle on-premise ad Amazon Aurora edizione compatibile con PostgreSQL.
- **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 su Amazon Relational Database Service (Amazon RDS) per Oracle nel 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: esegui la migrazione del 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 nel 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. Questo scenario di migrazione è specifico di VMware Cloud on AWS, che supporta la compatibilità delle macchine virtuali (VM) e la portabilità del carico di lavoro tra l'ambiente locale e AWS. È possibile utilizzare le tecnologie VMware Cloud Foundation dai data center on-premise durante la migrazione dell'infrastruttura a VMware Cloud su AWS. Esempio: trasferisci l'hypervisor che ospita il database Oracle su VMware Cloud on. 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 effettuare la migrazione.

- Ritirare: disattiva o rimuovi le applicazioni che non sono più necessarie nell'ambiente di origine.

## A

### 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 di 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.

### 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 di frequente 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 interrompere 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.

### distribuzioni blu/verdi

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, vedere l'indicatore [Implementate break-glass procedures](#) nella guida Well-Architected AWS .

## 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.

## crittografia lato client

Crittografia dei dati a livello locale, prima che il destinatario 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 sul blog](#) AWS Cloud Enterprise Strategy.

## cloud computing

La tecnologia cloud generalmente utilizzata per l'archiviazione remota di dati e la gestione dei dispositivi IoT. Il cloud computing è generalmente connesso 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 al AWS cloud:

- 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
- Reinvenzione: ottimizzazione di prodotti e servizi e innovazione nel cloud

Queste fasi sono state definite da Stephen Orban nel post sul blog The [Journey Toward Cloud-First & the Stages of Adoption on the](#) AWS Cloud Enterprise Strategy. [Per informazioni su come si relazionano alla 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 AWS CodeCommit. Ogni versione del codice è denominata ramo. In una struttura a microservizi, ogni repository è dedicato a una singola funzionalità. Una singola pipeline CI/CD 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, AWS Panorama offre dispositivi che aggiungono CV alle reti di telecamere locali e Amazon SageMaker 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 continua (continuous integration and continuous delivery, CI/CD)

Il processo di automazione delle fasi di origine, creazione, test, gestione temporanea e produzione del processo di rilascio del software. Il processo CI/CD è comunemente descritto come una pipeline. CI/CD può aiutare ad automatizzare i processi, migliorare la produttività, migliorare

la qualità del codice e velocizzare le distribuzioni. 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.

## defense-in-depth

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 defense-in-depth approccio potrebbe combinare l'autenticazione a più fattori, la segmentazione della rete e la 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 Workloads 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 il modello del fico strangolatore (Strangler Fig), consulta la sezione [Modernizzazione incrementale dei servizi Web Microsoft ASP.NET \(ASMX\) legacy utilizzando container e il Gateway Amazon API](#).

## 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](#).

### 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.

### 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. I sistemi big-endian memorizzano per primo il byte più importante. I sistemi little-endian memorizzano per primo 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 pipeline CI/CD, l'ambiente di produzione è l'ultimo ambiente di implementazione.
- 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 riproduzione 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 epopee 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 in uno [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, vedere [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.

## 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.

## G

### 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](#).

## 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 GuardDuty AWS Trusted Advisor, Amazon Inspector e controlli personalizzati AWS Lambda .

# 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.

## 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, come 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

## IaC

Considera [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.

I

## 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 in Well-Architected AWS Framework](#).

## 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 la sezione [Interpretabilità dei modelli 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](#).

### 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](#).

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](#).

## 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, consulta [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 machine-to-machine \(M2M\) leggero, basato sul modello di pubblicazione/sottoscrizione, 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

Team interfunzionali 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 al 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 nel 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, consulta la sezione [Valutazione della preparazione alla modernizzazione per le applicazioni nel 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 machine-to-machine (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 Well-Architected AWS Framework.

## tecnologia operativa (OT)

Sistemi hardware e software che interagiscono con l'ambiente fisico per controllare le operazioni, le apparecchiature e le 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 tutti i bucket S3 in generale Regioni AWS, la crittografia lato server con AWS KMS (SSE-KMS) e le richieste dinamiche e dirette al bucket S3.

PUT 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.

O

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. Per ulteriori informazioni, consulta la sezione [Abilitazione della persistenza dei dati nei microservizi](#).

#### 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 all'ingegneria dei sistemi che tiene conto della privacy durante l'intero processo di progettazione.

#### 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.

## 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.

## pubblica/iscriviti (pub/sub)

Un pattern 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 \(RACI\)](#).

### 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. Ciò determina quella che viene considerata 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.](#)

## Regione

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](#).

## 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 l'obiettivo del punto [di ripristino](#).

## RTO

Vedi l'[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 AWS Management Console 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 AWS Secrets Manager, 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 Secret](#) nella documentazione di Secrets Manager.

## 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](#).

## 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](#).

## split-and-seed modello

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 [Modernizzazione incrementale dei servizi Web legacy di Microsoft ASP.NET \(ASMX\) mediante container e Gateway Amazon API](#).

## 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.

# T

## tags

Coppie chiave-valore 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.

## 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, preproduzione 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. Per ulteriori informazioni, consulta la guida [Quantificazione dell'incertezza nei sistemi di deep learning](#).

### 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 query 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.

### 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.