



Guida per gli sviluppatori

# Servizio di posizione Amazon



# Servizio di posizione Amazon: Guida per gli sviluppatori

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

Benvenuto .....	1
Cos'è Amazon Location Service? .....	1
Funzionalità principali .....	2
Servizi correlati .....	3
Avvio rapido .....	5
Creazione di un'app Web .....	5
Creare risorse .....	6
Configurazione dell'autenticazione .....	7
Creazione di HTML .....	9
Aggiungere la mappa .....	12
Aggiungere una ricerca .....	16
Applicazione finale .....	21
Cosa c'è dopo .....	26
Creazione di un'app per Android .....	26
Creazione di risorse Amazon Location per la tua app .....	26
Configurazione dell'autenticazione .....	28
Crea l'app .....	31
Aggiungi la mappa .....	32
Aggiungi ricerca .....	35
Aggiungi tracciamento .....	45
Cosa c'è dopo .....	54
Creazione di un'app per iOS .....	55
Creare risorse .....	55
Configurazione dell'autenticazione .....	57
Creazione dell'app .....	59
Codice iniziale .....	60
Aggiungere una mappa .....	63
Aggiungi ricerca .....	67
Aggiungi tracciamento .....	69
Cosa c'è dopo .....	81
Concetti di localizzazione Amazon .....	82
Panoramica .....	83
Mappe .....	84
Stili di mappa .....	84

Opinioni politiche .....	85
Livelli personalizzati .....	86
Rendering della mappa .....	86
Terminologia delle mappe .....	87
Ricerca di luoghi .....	89
Concetti di geocodifica .....	90
Risultati della ricerca .....	91
Molteplici risultati e rilevanza .....	91
Indirizza i risultati .....	92
Memorizzazione dei risultati della geocodifica .....	94
Terminologia relativa ai luoghi .....	94
Route .....	95
Risorse per il calcolatore del percorso .....	96
Calcolo di un percorso .....	96
Pianificazione dei percorsi .....	98
Terminologia del percorso .....	99
Geofence e tracker .....	100
Geofence .....	100
Localizzatori .....	102
Terminologia Geofence .....	107
Terminologia del tracker .....	108
Casi di utilizzo comune .....	109
Applicazioni di geomarketing e coinvolgimento degli utenti .....	110
Applicazioni di tracciamento delle risorse .....	112
Applicazioni di distribuzione .....	113
Fornitori di dati .....	115
Copertura e funzionalità dei fornitori di dati .....	115
Stili di mappa .....	117
Ulteriori dettagli .....	117
Esri .....	117
GrabMaps .....	126
Tecnologie HERE .....	131
Open Data (Apri dati) .....	138
Funzionalità per fornitore di dati .....	148
Termini di utilizzo e attribuzione dei dati .....	153
Regioni ed endpoint .....	153

Regioni .....	154
Endpoints .....	156
Endpoint operativi tramite API .....	156
Quote del servizio .....	158
Gestione delle quote del servizio Amazon Location .....	180
Sviluppo con Amazon Location .....	182
Scenari e casi d'uso .....	182
SDK e strumenti .....	183
SDK per lingua .....	184
MapLibre .....	188
SDK di localizzazione Amazon .....	193
API di localizzazione Amazon .....	217
Utilizzo di Amazon Location con un AWS SDK .....	218
Aggiornamenti dei messaggi di errore .....	218
Esempi di codice .....	252
Sito demo di Amazon Location .....	253
Tutorial: avvio rapido .....	254
Tutorial: arricchimento del database .....	255
Esempio: app Explore .....	255
Esempio: assegnare uno stile a una mappa .....	256
Esempio: disegna dei marcatori .....	256
Esempio: disegna punti raggruppati .....	257
Esempio: disegnare un poligono .....	257
Esempio: modifica della lingua della mappa .....	258
Blog: notifiche sui tempi di consegna stimati .....	259
Esempio: aggiornamenti della posizione dello stream .....	260
Esempio: applicazione mobile di geofencing e tracciamento .....	260
Come usare Amazon Location .....	262
Prerequisiti dell'account .....	263
Registrati per un Account AWS .....	263
Crea un utente con accesso amministrativo .....	264
Concedi l'accesso ad Amazon Location Service .....	265
Utilizzo delle mappe .....	267
Prerequisiti .....	268
Visualizzazione delle mappe .....	271
Disegnare su una mappa .....	326

Impostazioni, estensioni per una mappa .....	326
Gestione delle risorse cartografiche .....	328
Ricerca di luoghi .....	331
Prerequisiti .....	332
Geocodifica .....	336
Geocodifica inversa .....	343
Completamento automatico .....	347
Utilizzo degli ID dei luoghi .....	354
Categorie e filtri .....	356
Tutorial: arricchimento del database .....	361
Gestione delle risorse relative all'indice dei luoghi .....	376
Calcolo dei percorsi .....	379
Prerequisiti .....	380
Calcola il percorso .....	384
Pianificazione del percorso .....	388
Posizioni non situate su una strada .....	394
Ora di partenza .....	396
Modalità di viaggio .....	397
Gestione delle risorse relative agli itinerari .....	399
Geofencing e tracciamento .....	402
Fase 1: Aggiungere geofence .....	404
Fase 2: Inizia il tracciamento .....	411
Passaggio 3: collega un tracker a una raccolta di geofence .....	425
Fase 4: Valutare le posizioni dei dispositivi rispetto ai geofence .....	427
Verifica le posizioni dei dispositivi .....	430
Reagire agli eventi con EventBridge .....	432
Monitoraggio tramite AWS IoT e MQTT .....	437
Gestione delle risorse di geofence .....	445
Gestione delle risorse del tracker .....	453
Esempio di applicazione mobile Geofencing and Tracking .....	458
Tagging delle risorse .....	476
Restrizioni .....	477
Concedi il permesso di taggare .....	478
Aggiungi un tag a una risorsa .....	479
Tieni traccia dei costi per tag .....	479
Controlla l'accesso alle risorse mediante i tag .....	480

Ulteriori informazioni .....	481
Concedi l'accesso ad Amazon Location .....	481
Utilizzo delle chiavi API .....	482
Utilizzo di Amazon Cognito .....	488
Monitoraggio di Amazon Location Service .....	499
Monitoraggio con CloudWatch .....	500
Utilizzo CloudTrail con Amazon Location .....	505
Utilizzare AWS CloudFormation per creare risorse .....	509
Ubicazione e AWS CloudFormation modelli Amazon .....	510
Ulteriori informazioni su AWS CloudFormation .....	510
Sicurezza .....	511
Protezione dei dati .....	512
Privacy dei dati .....	513
Conservazione dei dati .....	513
Crittografia dei dati inattivi .....	513
Crittografia dei dati in transito .....	527
Identity and Access Management .....	527
Destinatari .....	527
Autenticazione con identità .....	528
Gestione dell'accesso con policy .....	532
Come funziona Amazon Location Service con IAM .....	534
Come funziona Amazon Location Service con utenti non autenticati .....	542
Esempi di policy basate su identità .....	543
Risoluzione dei problemi .....	556
Risposta agli incidenti .....	558
Registrazione e monitoraggio .....	558
Convalida della conformità .....	559
Resilienza .....	560
Sicurezza dell'infrastruttura .....	560
Analisi della configurazione e delle vulnerabilità .....	561
Prevenzione del "confused deputy" .....	561
Best practice di sicurezza .....	562
Le migliori pratiche da Detective .....	562
Le migliori pratiche di prevenzione .....	563
Best practice .....	563
Sicurezza .....	564

---

Gestione delle risorse .....	564
Gestione di costi e fatturazione .....	565
Quote e utilizzo .....	565
Cronologia dei documenti .....	567
Glossario AWS .....	577
.....	dlxxviii



# Benvenuto su Amazon Location Service

Benvenuto nella Guida per gli sviluppatori di Amazon Location Service.

I seguenti argomenti possono aiutarti a iniziare a leggere la documentazione, in base a ciò che stai cercando di fare.

Ottieni una panoramica di Amazon Location

- Scopri i [concetti in Amazon Location](#).
- Approfondisci le funzionalità nel [Come usare Amazon Location Service](#) capitolo.
- Guarda le app demo nel [sito demo di Amazon Location](#).
- Se ne possiedi già uno Account AWS, puoi utilizzare la [console Amazon Location Service](#) per esplorare in prima persona la funzionalità.

Usa Amazon Location come sviluppatore

- Crea la tua prima app con [Avvio rapido](#).
- Scopri come funzionano le varie funzionalità di Amazon Location Service nel [Come usare Amazon Location Service](#) capitolo.
- Scopri gli SDK e gli strumenti a tua disposizione nel [Sviluppo con Amazon Location](#) capitolo.
- Guarda [esempi di codice e tutorial](#) che puoi usare nelle tue app. Puoi anche visitare la [pagina degli esempi](#) del sito demo di Amazon Location per trovare esempi, filtrabili per funzionalità, lingua o piattaforma.
- Ottieni informazioni sulle API di Amazon Location nella [guida di riferimento delle API](#).

## Cos'è Amazon Location Service?

Amazon Location Service consente di aggiungere dati e funzionalità sulla posizione alle applicazioni, tra cui mappe, punti di interesse, geocodifica, routing, geofence e tracciamento. Amazon Location fornisce servizi basati sulla posizione (LBS) utilizzando dati di alta qualità provenienti da fornitori globali affidabili come Esri, Grab e HERE. Con dati convenienti, funzionalità di tracciamento e geofencing e metriche integrate per il monitoraggio della salute, puoi creare applicazioni sofisticate abilitate alla localizzazione.

Con Amazon Location, mantieni il controllo dei dati della tua organizzazione. Amazon Location rende anonime tutte le richieste inviate ai fornitori di dati rimuovendo i metadati dei clienti e le informazioni sull'account. Inoltre, le informazioni sensibili di tracciamento e geofencing sulla posizione, come

strutture, risorse e ubicazioni del personale, non escono affatto dal tuo account. AWS Ciò consente di proteggere le informazioni sensibili da terze parti, proteggere la privacy degli utenti e ridurre i rischi per la sicurezza delle applicazioni. Con Amazon Location, Amazon e terze parti non hanno il diritto di vendere i tuoi dati o utilizzarli per scopi pubblicitari.

Amazon Location è completamente integrato con servizi come Amazon AWS CloudTrail CloudWatch EventBridge, Amazon e AWS Identity and Access Management (IAM). Amazon Location semplifica il flusso di lavoro di sviluppo con l'integrazione dei dati e accelera la produzione delle app con funzionalità integrate di monitoraggio, sicurezza e conformità.

Per i punti salienti, i dettagli del prodotto e i prezzi, consulta la pagina di assistenza per [Amazon Location Service](#).

## Caratteristiche principali di Amazon Location

Amazon Location offre le seguenti funzionalità:

### Mappe

Amazon Location Service Maps ti consente di visualizzare le informazioni sulla posizione ed è alla base di molte funzionalità di servizi basati sulla posizione. Amazon Location Service fornisce riquadri di mappe di stili diversi provenienti da fornitori globali di dati di localizzazione Esri, Grab e HERE, oltre a mappe Open data.

### Luoghi

Amazon Location Service Places ti consente di integrare la funzionalità di ricerca nella tua applicazione, convertire gli indirizzi in coordinate geografiche di latitudine e longitudine (geocodifica) e convertire una coordinata in un indirizzo stradale (geocodifica inversa). Amazon Location Service raccoglie dati geospaziali di alta qualità da Esri, Grab e HERE per supportare le funzioni di Places.

### Routing

Amazon Location Service Routes ti consente di trovare percorsi e stimare il tempo di viaggio in base alle informazioni sulla up-to-date carreggiata e sul traffico in tempo reale. Crea funzionalità che consentano alla tua applicazione di richiedere il tempo di viaggio, la distanza e le indicazioni stradali tra due località qualsiasi. Calcola il tempo e la distanza per una matrice di percorsi da utilizzare nella pianificazione del percorso.

## Geofencing

Amazon Location Service Geofences ti consente di dare alla tua applicazione la capacità di rilevare e agire quando un dispositivo entra o esce da un confine geografico definito noto come geofence. Invia automaticamente un evento di entrata o uscita ad Amazon EventBridge quando viene rilevata una violazione del geofence. Ciò consente di avviare azioni a valle come l'invio di una notifica a un obiettivo.

## Tracker

Amazon Location Service Trackers ti consente di recuperare la posizione attuale e storica dei dispositivi che eseguono la tua applicazione abilitata al tracciamento. Puoi anche collegare i tracker ai geofence di Amazon Location Service per valutare automaticamente gli aggiornamenti di posizione dai tuoi dispositivi rispetto ai geofence. I tracker possono aiutarti a ridurre i costi filtrando gli aggiornamenti di posizione che non sono stati spostati prima di essere archiviati o valutati in base ai geofence.

Quando utilizzi i tracker, le informazioni sensibili sulla posizione sui dispositivi monitorati non escono dal tuo account. AWS Questo aiuta a proteggere le informazioni sensibili da terze parti, a proteggere la privacy degli utenti e a ridurre i rischi per la sicurezza.

## Servizi che puoi utilizzare con Amazon Location

Utilizza i seguenti servizi insieme ad Amazon Location Service.

### Monitoraggio e gestione integrati

Amazon Location Service è integrato con Amazon CloudWatch e Amazon EventBridge per un monitoraggio e una gestione dei dati efficienti: AWS CloudTrail

- Amazon CloudWatch: visualizza i parametri sull'utilizzo e lo stato del servizio, tra cui richieste, latenza, errori e log. Per ulteriori informazioni, consulta [the section called "Monitoraggio con CloudWatch"](#).
- AWS CloudTrail— Registra e monitora le tue chiamate API, che includono le azioni intraprese da un utente, un ruolo o un servizio AWS. Per ulteriori informazioni, consulta [the section called "Utilizzo CloudTrail con Amazon Location"](#).

- Amazon EventBridge: abilita un'architettura applicativa basata sugli eventi in modo da poter utilizzare AWS Lambda le funzioni per attivare altre parti dell'applicazione e dei flussi di lavoro. Per ulteriori informazioni, consulta [the section called “Reagire agli eventi con EventBridge”](#).

## Strumenti per sviluppatori

Amazon Location Service offre agli sviluppatori una varietà di strumenti per creare applicazioni abilitate alla localizzazione. Questi includono gli AWS SDK standard, gli SDK per dispositivi mobili e web e il codice di esempio per combinarli con librerie open source come MapLibre. Usa la [console Amazon Location Service](#) per conoscere le risorse e iniziare a usare uno strumento di apprendimento visivo e interattivo.

# Avvio rapido con Amazon Location Service

Il modo più efficiente per iniziare a usare Amazon Location Service è utilizzare la [console Amazon Location](#). Puoi creare e gestire le tue risorse e provare la funzionalità Amazon Location utilizzando [la pagina Esplora](#).

## Note

Per utilizzare la console Amazon Location Service o seguire il resto di questo tutorial, è necessario prima completare il modulo [Prerequisiti per l'utilizzo di Amazon Location Service](#), inclusa la creazione di un AWS account e l'autorizzazione all'accesso ad Amazon Location.

Per iniziare a conoscere le API di Amazon Location, usa il seguente tutorial per creare una semplice applicazione che mostri una mappa interattiva e utilizzi la funzionalità di ricerca. Esistono tre versioni del tutorial: una mostra come creare una semplice pagina Web utilizzando JavaScript, la seconda mostra lo stesso per un'applicazione Android che utilizza Kotlin e la terza mostra lo stesso per un'applicazione iOS che utilizza Swift.

## Argomenti

- [Creazione di un'app Web](#)
- [Creazione di un'app per Android](#)
- [Creazione di un'app per iOS](#)

## Creazione di un'app Web

In questa sezione, creerai una pagina web statica con una mappa e la possibilità di effettuare ricerche in una località. Per prima cosa creerai le tue risorse Amazon Location e creerai una chiave API per la tua applicazione.

## Argomenti

- [Creazione di risorse Amazon Location per la tua app](#)
- [Configurazione dell'autenticazione per l'applicazione](#)
- [Creazione del codice HTML per l'applicazione](#)
- [Aggiungere una mappa interattiva all'applicazione](#)

- [Aggiungere la ricerca all'applicazione](#)
- [Visualizzazione dell'applicazione finale](#)
- [Cosa c'è dopo](#)

## Creazione di risorse Amazon Location per la tua app

Se non li possiedi già, devi creare le risorse Amazon Location che l'applicazione utilizzerà. Qui, crei una risorsa cartografica per visualizzare le mappe nella tua applicazione e un indice dei luoghi per cercare le posizioni sulla mappa.

Per aggiungere risorse di localizzazione all'applicazione

1. Scegli lo stile di mappa che desideri utilizzare.
  - a. Nella console Amazon Location, nella pagina [Mappe](#), scegli Crea mappa per visualizzare in anteprima gli stili delle mappe.
  - b. Aggiungi un nome e una descrizione per la nuova risorsa cartografica. Prendi nota del nome che usi per la risorsa della mappa. Ne avrai bisogno quando creerai il tuo file di script più avanti nel tutorial.
  - c. Scegli una mappa.

### Note

La scelta di uno stile di mappa consente anche di scegliere il fornitore di dati cartografici da utilizzare. Se la tua applicazione monitora o indirizza le risorse che utilizzi nella tua attività, come veicoli per le consegne o dipendenti, puoi utilizzare HERE solo come fornitore di geolocalizzazione. [Per ulteriori informazioni, consulta la sezione 82 dei termini di servizio. AWS](#)

- d. Accetta i Termini e condizioni di Amazon Location, quindi scegli Crea mappa. Puoi interagire con la mappa che hai scelto: ingrandisci, rimpicciolisci o fai una panoramica in qualsiasi direzione.
  - e. Prendi nota dell'Amazon Resource Name (ARN) visualizzato per la tua nuova risorsa cartografica. Lo utilizzerai per creare l'autenticazione corretta più avanti in questo tutorial.
2. Scegli l'indice dei luoghi che desideri utilizzare.

- a. Nella console Amazon Location, nella pagina degli [indici dei luoghi](#), scegli Crea indice dei luoghi.
- b. Aggiungi un nome e una descrizione per la nuova risorsa relativa all'indice dei luoghi. Prendi nota del nome che usi per la risorsa dell'indice dei luoghi. Ne avrai bisogno quando creerai il tuo file di script più avanti nel tutorial.
- c. Scegli un fornitore di dati.

#### Note

Nella maggior parte dei casi, scegli il fornitore di dati che corrisponde al fornitore di mappe che hai già scelto. Questo aiuta a garantire che le ricerche corrispondano alle mappe.

Se la tua applicazione monitora o indirizza le risorse che utilizzi nella tua attività, come veicoli per le consegne o dipendenti, puoi utilizzare HERE solo come provider di geolocalizzazione. [Per ulteriori informazioni, consulta la sezione 82 dei termini di servizio. AWS](#)

- d. Scegli l'opzione di archiviazione dei dati. Per questo tutorial, i risultati non vengono memorizzati, quindi puoi scegliere No, solo monouso.
- e. Accetta i Termini e condizioni di Amazon Location, quindi scegli Crea indice dei luoghi.
- f. Prendi nota dell'ARN mostrato per la tua nuova risorsa dell'indice dei luoghi. Lo userai per creare l'autenticazione corretta nella prossima sezione di questo tutorial.

## Configurazione dell'autenticazione per l'applicazione

L'applicazione creata in questo tutorial ha un utilizzo anonimo, il che significa che gli utenti non devono accedere AWS per utilizzare l'applicazione. Tuttavia, per impostazione predefinita, le API di Amazon Location Service richiedono l'autenticazione per essere utilizzate. Puoi utilizzare Amazon Cognito o le chiavi API per fornire l'autenticazione e l'autorizzazione agli utenti anonimi. In questo tutorial, creerai chiavi API da utilizzare nell'applicazione di esempio.

#### Note

Per ulteriori informazioni sull'utilizzo delle chiavi API o di Amazon Cognito con Amazon Location Service, consulta. [Concessione dell'accesso ad Amazon Location Service](#)


Per configurare l'autenticazione per la tua applicazione

1. Vai alla [console Amazon Location](#) e scegli le chiavi API dal menu a sinistra.
2. Scegli Crea chiave API.

 Important

La chiave API che crei deve trovarsi nella stessa Account AWS AWS regione delle risorse di Amazon Location Service che hai creato nella sezione precedente.

3. Nella pagina Crea chiave API, inserisci le seguenti informazioni.
  - Nome: un nome per la tua chiave API, ad esempioMyWebAppKey.
  - Risorse: scegli le risorse Amazon Location Map e Place index che hai creato nella sezione precedente. Puoi aggiungere più di una risorsa selezionando Aggiungi risorsa. Ciò consentirà di utilizzare la chiave API con tali risorse.
  - Azioni: specifica le azioni che desideri autorizzare con questa chiave API. Devi selezionare almeno geo: GetMap \* e geo: SearchPlaceIndexForPosition in modo che il tutorial funzioni come previsto.
  - Facoltativamente, puoi aggiungere una descrizione, una data di scadenza o dei tag alla tua chiave API. Puoi anche aggiungere un referente (ad esempio\*.example.com), per limitare l'utilizzo della chiave solo da un determinato dominio. Ciò significa che il tutorial funzionerà solo da quel dominio.

 Note

Si consiglia di proteggere l'utilizzo della chiave API impostando un orario di scadenza o un referente, se non entrambi.

4. Scegli Crea chiave API per creare la chiave API.
5. Scegli Mostra chiave API e copia il valore della chiave per utilizzarlo più avanti nel tutorial. Sarà nel modulov1.public.a1b2c3d4....



**⚠ Important**

Avrai bisogno di questa chiave per scrivere il codice per la tua applicazione più avanti in questo tutorial.

## Creazione del codice HTML per l'applicazione

In questo tutorial, creerai una pagina HTML statica che incorpora una mappa e consente all'utente di trovare cosa si trova in una posizione sulla mappa. L'app sarà composta da tre file: un file HTML e un file CSS per la pagina web e un file JavaScript (.js) per il codice che crea la mappa e risponde alle interazioni dell'utente e agli eventi della mappa.

Innanzitutto, creiamo il framework HTML e CSS che verrà utilizzato per l'applicazione. Questa sarà una pagina semplice con un `<div>` elemento per contenere il contenitore della mappa e un `<pre>` elemento per mostrare le risposte JSON alle tue domande.

Per creare il codice HTML per l'applicazione di avvio rapido

1. Crea un nuovo file denominato `quickstart.html`.
2. Modifica il file nell'editor di testo o nell'ambiente che preferisci. Aggiungi il seguente codice HTML al file.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Quick start tutorial</title>

    <!-- Styles -->
    <link href="main.css" rel="stylesheet" />
  </head>

  <body>
    <header>
      <h1>Quick start tutorial</h1>
    </header>
    <main>
      <div id="map"></div>
      <aside>
```

```
<h2>JSON Response</h2>
<pre id="response"></pre>
</aside>
</main>
<footer>This is a simple Amazon Location Service app. Pan and zoom. Click to
see details about entities close to a point.</footer>

</body>
</html>
```

Questo codice HTML contiene un puntatore al file CSS che verrà creato nel passaggio successivo, alcuni elementi segnaposto per l'applicazione e del testo esplicativo.

Ci sono due elementi segnaposto che utilizzerete più avanti in questo tutorial. Il primo è l'<div id="map">elemento, che manterrà il controllo della mappa. Il secondo è l'<pre id="response">elemento, che mostrerà i risultati della ricerca sulla mappa.

### 3. Salva il file.

Ora aggiungi il CSS per la pagina web. Questo imposterà lo stile del testo e degli elementi segnaposto per l'applicazione.

Per creare il CSS per l'applicazione ad avvio rapido

1. Create un nuovo file denominato `main.css`, nella stessa cartella del file `quickstart.html` creato nella procedura precedente.
2. Modifica il file nell'editor che desideri utilizzare. Aggiungi il seguente testo al file.

```
* {
  box-sizing: border-box;
  font-family: Arial, Helvetica, sans-serif;
}

body {
  margin: 0;
}

header {
  background: #000000;
  padding: 0.5rem;
}
```

```
h1 {
  margin: 0;
  text-align: center;
  font-size: 1.5rem;
  color: #ffffff;
}

main {
  display: flex;
  min-height: calc(100vh - 94px);
}

#map {
  flex: 1;
}

aside {
  overflow-y: auto;
  flex: 0 0 30%;
  max-height: calc(100vh - 94px);
  box-shadow: 0 1px 1px 0 #001c244d, 1px 1px 1px 0 #001c2426, -1px 1px 1px 0 #001c2426;
  background: #f9f9f9;
  padding: 1rem;
}

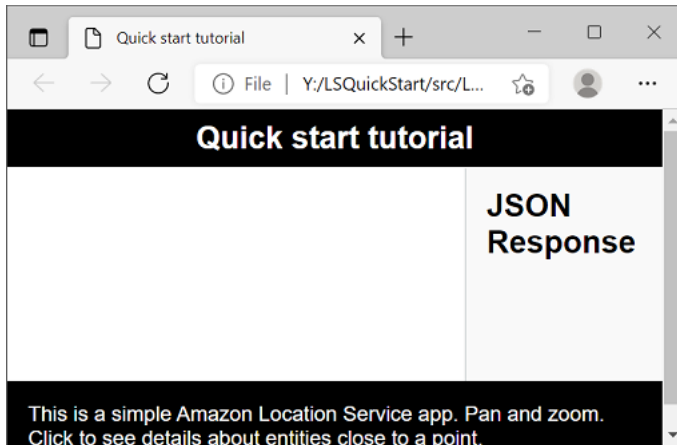
h2 {
  margin: 0;
}

pre {
  white-space: pre-wrap;
  font-family: monospace;
  color: #16191f;
}

footer {
  background: #000000;
  padding: 1rem;
  color: #ffffff;
}
```

Ciò imposta la mappa in modo che riempia lo spazio non utilizzato da nient'altro, imposta l'area delle nostre risposte in modo che occupi il 30% della larghezza dell'app e imposta il colore e gli stili per il titolo e il testo esplicativo.

3. Salvare il file.
4. Ora puoi visualizzare il `quickstart.html` file in un browser per vedere il layout dell'applicazione.



Successivamente, aggiungerai il controllo della mappa all'applicazione.

## Aggiungere una mappa interattiva all'applicazione

Ora che hai un framework e un segnaposto `div`, puoi aggiungere il controllo della mappa alla tua applicazione. Questo tutorial utilizza [MapLibre GL JS](#) come controllo della mappa, ottenendo dati da Amazon Location Service. Utilizzerai anche il [JavaScript Aiutante di autenticazione](#) per facilitare la firma delle chiamate alle API di Amazon Location con la tua chiave API.

Per aggiungere una mappa interattiva alla tua applicazione

1. Apri il `quickstart.html` file che hai creato nella sezione precedente.
2. Aggiungete i riferimenti alle librerie necessarie e al file di script che creerete. Le modifiche da apportare sono mostrate **ingreen**.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Quick start tutorial</title>
```

```
<!-- Styles -->
<link href="https://unpkg.com/maplibre-gl@3.x/dist/maplibre-gl.css"
rel="stylesheet" />
<link href="main.css" rel="stylesheet" />
</head>

<body>
  ...
  <footer>This is a simple Amazon Location Service app. Pan and zoom. Click to
see details about entities close to a point.</footer>

  <!-- JavaScript dependencies -->
  <script src="https://unpkg.com/maplibre-gl@3.x/dist/maplibre-gl.js"></script>
  <script src="https://unpkg.com/@aws/amazon-location-client@1.x/dist/
amazonLocationClient.js"></script>
  <script src="https://unpkg.com/@aws/amazon-location-utilities-auth-helper@1.x/
dist/amazonLocationAuthHelper.js"></script>

  <!-- JavaScript for the app -->
  <script src="main.js"></script>
</body>
</html>
```

Ciò aggiunge le seguenti dipendenze all'app:

- MapLibre VAI JS. Questa libreria e questo foglio di stile includono un controllo cartografico che mostra i riquadri della mappa e include elementi interattivi, come panoramica e zoom. Il controllo consente anche estensioni, come disegnare caratteristiche personalizzate sulla mappa.
- Client Amazon Location. Ciò fornisce interfacce per la funzionalità Amazon Location necessaria per ottenere dati cartografici e cercare luoghi sulla mappa. Il client Amazon Location si basa sull'AWS SDK per la JavaScript versione 3.
- Amazon Location Authentication Helper. Ciò fornisce funzioni utili per l'autenticazione di Amazon Location Service con chiavi API o Amazon Cognito.

Questo passaggio aggiunge anche un riferimento `main.js`, che verrà creato in seguito.

### 3. Salvare il file `quickstart.html`.

4. Create un nuovo file chiamato `main.js` nella stessa cartella dei file HTML e CSS e apritelo per modificarlo.
5. Aggiungi il seguente script al tuo file. Il testo in *rosso* deve essere sostituito con il valore della chiave API, il nome della risorsa della mappa e il nome della risorsa del luogo che hai creato in precedenza, nonché l'identificatore della regione (ad esempio `us-east-1`).

```
// Amazon Location Service resource names:
const mapName = "explore.map";
const placesName = "explore.place";
const region = "your_region";
const apiKey = "v1.public.a1b2c3d4..."

// Initialize a map
async function initializeMap() {
  const mlglMap = new maplibregl.Map({
    container: "map", // HTML element ID of map element
    center: [-77.03674, 38.891602], // Initial map centerpoint
    zoom: 16, // Initial map zoom
    style: 'https://maps.geo.${region}.amazonaws.com/maps/v0/maps/${mapName}/style-
descriptor?key=${apiKey}', // Defines the appearance of the map and authenticates
    using an API key
  });

  // Add navigation control to the top left of the map
  mlglMap.addControl(new maplibregl.NavigationControl(), "top-left");

  return mlglMap;
}

async function main() {
  // Initialize map and Amazon Location SDK client:
  const map = await initializeMap();
}

main();
```

Questo codice configura le risorse di Amazon Location, quindi configura e inizializza un controllo mappa MapLibre GL JS e lo inserisce nel tuo `<div>` elemento con l'id. `map`

La `initializeMap()` funzione è importante da capire. Crea un nuovo controllo della MapLibre mappa (chiamato `mglMap` localmente, ma chiamato `map` nel resto del codice) che viene utilizzato per eseguire il rendering della mappa nell'applicazione.

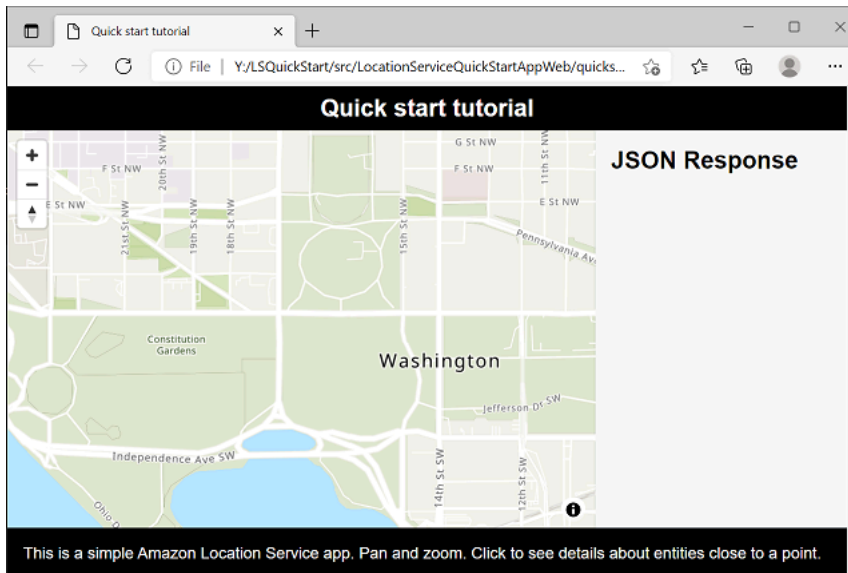
```
// Initialize the map
const mglMap = new maplibregl.Map({
  container: "map", // HTML element ID of map element
  center: [-77.03674, 38.891602], // Initial map centerpoint
  zoom: 16, // Initial map zoom
  style: 'https://maps.geo.${region}.amazonaws.com/maps/v0/maps/${mapName}/style-
descriptor?key=${apiKey}', // Defines the appearance of the map and authenticates
  using an API key
});
```

Quando si crea un nuovo controllo di MapLibre mappa, i parametri passati indicano lo stato iniziale del controllo della mappa. Qui impostiamo i seguenti parametri.

- Contenitore HTML, che utilizza l'elemento `map` div nel nostro codice HTML.
  - Il centro iniziale della mappa di un punto a Washington, DC.
  - Il livello di zoom a 16 (ingrandito a livello di quartiere o blocco).
  - Lo stile da usare per la mappa, che fornisce MapLibre un URL da utilizzare per ottenere i riquadri della mappa e altre informazioni per il rendering della mappa. Nota che questo URL include la tua chiave API per l'autenticazione.
6. Salva il JavaScript file e aprilo con un browser. Ora hai una mappa sulla pagina, in cui puoi usare le azioni di panoramica e zoom.

#### Note

Puoi usare questa app per vedere come si comporta il controllo della MapLibre mappa. Puoi provare a usare `Ctrl` o `Shift` mentre usi un'operazione di trascinamento, per vedere altri modi di interagire con la mappa. Tutte queste funzionalità sono personalizzabili.



La tua app è quasi completa. Nella sezione successiva, gestirai la scelta di una posizione sulla mappa e mostrerai l'indirizzo della posizione scelta. Mostrerai anche il codice JSON risultante sulla pagina, per vedere i risultati completi.

## Aggiungere la ricerca all'applicazione

L'ultimo passaggio dell'applicazione consiste nell'aggiungere la ricerca sulla mappa. In questo caso, aggiungerai una ricerca con geocodifica inversa, in cui troverai gli elementi in una posizione.

### Note

Amazon Location Service offre anche la possibilità di cercare per nome o indirizzo per trovare le posizioni dei luoghi sulla mappa.

Per aggiungere funzionalità di ricerca alla tua applicazione

1. Apri il `main.js` file che hai creato nella sezione precedente.
2. Modificate la `main` funzione, come illustrato. Le modifiche da apportare sono mostrate **ingreen**.

```
async function main() {  
  // Create an authentication helper instance using an API key  
  const authHelper = await amazonLocationAuthHelper.withAPIKey(apiKey);
```



```
// Initialize map and Amazon Location SDK client:
const map = await initializeMap();

const client = new amazonLocationClient.LocationClient({
  region,
  ...authHelper.getLocationClientConfig(), // Provides configuration required to
make requests to Amazon Location
});

// On mouse click, display marker and get results:
map.on("click", async function (e) {
  // Set up parameters for search call
  let params = {
    IndexName: placesName,
    Position: [e.lngLat.lng, e.lngLat.lat],
    Language: "en",
    MaxResults: "5",
  };

  // Set up command to search for results around clicked point
  const searchCommand = new
amazonLocationClient.SearchPlaceIndexForPositionCommand(params);

  try {
    // Make request to search for results around clicked point
    const data = await client.send(searchCommand);

    // Write JSON response data to HTML
    document.querySelector("#response").textContent = JSON.stringify(data,
undefined, 2);

    // Display place label in an alert box
    alert(data.Results[0].Place.Label);
  } catch (error) {
    // Write JSON response error to HTML
    document.querySelector("#response").textContent = JSON.stringify(error,
undefined, 2);

    // Display error in an alert box
    alert("There was an error searching.");
  }
}
```

```
});  
}
```

Questo codice inizia configurando l'helper di autenticazione Amazon Location per utilizzare la tua chiave API.

```
const authHelper = await amazonLocationAuthHelper.withAPIKey(apiKey);
```

Quindi utilizza l'helper di autenticazione e la regione che stai utilizzando per creare un nuovo client Amazon Location.

```
const client = new amazonLocationClient.LocationClient({  
  region,  
  ...authHelper.getLocationClientConfig(),  
});
```

Successivamente, il codice risponde all'utente che sceglie un punto sul controllo della mappa. Lo fa rilevando un evento MapLibre fornito per. `click`

```
map.on("click", async function(e) {  
  ...  
});
```

L' `MapLibre` `click` evento fornisce parametri che includono la latitudine e la longitudine scelte dall'utente (`e.lngLat`). All'interno dell'evento, il codice crea il comando `searchPlaceIndexForPositionCommand` per trovare le entità alla latitudine e alla longitudine specificate.

```
// Set up parameters for search call  
let params = {  
  IndexName: placesName,  
  Position: [e.lngLat.lng, e.lngLat.lat],  
  Language: "en",  
  MaxResults: "5"  
};  
  
// Set up command to search for results around clicked point  
const searchCommand = new  
amazonLocationClient.SearchPlaceIndexForPositionCommand(params);
```

```
try {
  // Make request to search for results around clicked point
  const data = await client.send(searchCommand);
  ...
});
```

Qui, `IndexName` il nome della risorsa Place Index che hai creato in precedenza, la `Position` latitudine e la longitudine da cercare, `Language` è la lingua preferita per i risultati e indica ad `MaxResults` Amazon Location di restituire solo un massimo di cinque risultati.

Il codice rimanente verifica la presenza di un errore, quindi visualizza i risultati della ricerca nell'elemento chiamato `response` e mostra il risultato principale in una casella di avviso.

3. (Facoltativo) Se adesso salvate e aprite il `quickstart.html` file in un browser, scegliendo una posizione sulla mappa vi mostrerà il nome o l'indirizzo del luogo scelto.
4. Il passaggio finale dell'applicazione consiste nell'utilizzare la MapLibre funzionalità per aggiungere un marker nel punto selezionato dall'utente. Modificate la `main` funzione come segue. Le modifiche da apportare sono mostrate in **green**.

```
async function main() {
  // Create an authentication helper instance using an API key
  const authHelper = await amazonLocationAuthHelper.withAPIKey(apiKey);

  // Initialize map and Amazon Location SDK client
  const map = await initializeMap();
  const client = new amazonLocationClient.LocationClient({
    region,
    ...authHelper.getLocationClientConfig(), // Provides configuration required to
    make requests to Amazon Location
  });

  // Variable to hold marker that will be rendered on click
  let marker;

  // On mouse click, display marker and get results:
  map.on("click", async function (e) {
    // Remove any existing marker
    if (marker) {
      marker.remove();
    }

    // Render a marker on clicked point
```

```

marker = new maplibregl.Marker().setLngLat([e.lngLat.lng,
e.lngLat.lat]).addTo(map);

// Set up parameters for search call
let params = {
  IndexName: placesName,
  Position: [e.lngLat.lng, e.lngLat.lat],
  Language: "en",
  MaxResults: "5",
};

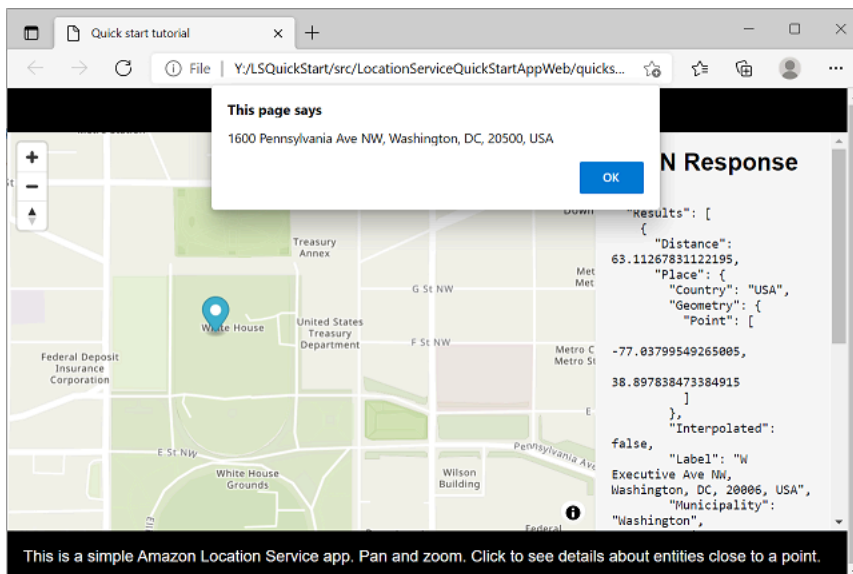
// Set up command to search for results around clicked point
const searchCommand = new
amazonLocationClient.SearchPlaceIndexForPositionCommand(params);

...

```

Questo codice dichiara una `marker` variabile, che viene compilata ogni volta che l'utente seleziona una posizione, mostrando dove è stata selezionata. Il marker viene renderizzato automaticamente dal controllo della mappa, una volta aggiunto alla mappa con `.addTo(map)`; Il codice verifica anche la presenza di un marker precedente e lo rimuove, in modo che sullo schermo sia presente un solo marker alla volta.

5. Salvate il `main.js` file e apritelo `quickstart.html` in un browser. È possibile eseguire la panoramica e lo zoom sulla mappa, come in precedenza, ma ora se si sceglie una posizione, verranno visualizzati i dettagli relativi alla posizione scelta.



L'applicazione di avvio rapido è completa. Questo tutorial ti ha mostrato come creare un'applicazione HTML statica che:

- Crea una mappa con cui gli utenti possono interagire.
- Gestisce un evento sulla mappa (`click`).
- Richiama un'API Amazon Location Service, in particolare per cercare sulla mappa di una posizione, utilizzando `searchPlaceIndexForPosition`.
- Utilizza il controllo della MapLibre mappa per aggiungere un marker.

## Visualizzazione dell'applicazione finale

Il codice sorgente finale per questa applicazione è incluso in questa sezione. Puoi trovare il progetto finale anche [su GitHub](#).

[Puoi anche trovare una versione dell'applicazione che utilizza Amazon Cognito anziché le chiavi API. GitHub](#)

### Overview

Seleziona ogni scheda per visualizzare il codice sorgente finale dei file in questo tutorial di avvio rapido.

I file sono:

- `quickstart.html`: il framework per l'applicazione, inclusi i supporti degli elementi HTML per la mappa e i risultati di ricerca.
- `main.css`: il foglio di stile per l'applicazione.
- `main.js`: lo script per l'applicazione che autentica l'utente, crea la mappa e cerca un evento. `click`

### quickstart.html

Il framework HTML per l'applicazione di avvio rapido.

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Quick start tutorial</title>
```

```
<!-- Styles -->
<link href="https://unpkg.com/maplibre-gl@3.x/dist/maplibre-gl.css"
rel="stylesheet" />
<link href="main.css" rel="stylesheet" />
</head>

<body>
  ...
  <footer>This is a simple Amazon Location Service app. Pan and zoom. Click to see
details about entities close to a point.</footer>

  <!-- JavaScript dependencies -->
  <script src="https://unpkg.com/maplibre-gl@3.x/dist/maplibre-gl.js"></script>
  <script src="https://unpkg.com/@aws/amazon-location-client@1.x/dist/
amazonLocationClient.js"></script>
  <script src="https://unpkg.com/@aws/amazon-location-utilities-auth-helper@1.x/
dist/amazonLocationAuthHelper.js"></script>

  <!-- JavaScript for the app -->
  <script src="main.js"></script>
</body>
</html>
```

## main.css

Il foglio di stile per l'applicazione di avvio rapido.

```
* {
  box-sizing: border-box;
  font-family: Arial, Helvetica, sans-serif;
}

body {
  margin: 0;
}

header {
  background: #000000;
  padding: 0.5rem;
}

h1 {
  margin: 0;
```

```
    text-align: center;
    font-size: 1.5rem;
    color: #ffffff;
  }

  main {
    display: flex;
    min-height: calc(100vh - 94px);
  }

  #map {
    flex: 1;
  }

  aside {
    overflow-y: auto;
    flex: 0 0 30%;
    max-height: calc(100vh - 94px);
    box-shadow: 0 1px 1px 0 #001c244d, 1px 1px 1px 0 #001c2426, -1px 1px 1px 0
    #001c2426;
    background: #f9f9f9;
    padding: 1rem;
  }

  h2 {
    margin: 0;
  }

  pre {
    white-space: pre-wrap;
    font-family: monospace;
    color: #16191f;
  }

  footer {
    background: #000000;
    padding: 1rem;
    color: #ffffff;
  }
}
```

## main.js

Il codice per l'applicazione di avvio rapido. Il testo in **rosso** deve essere sostituito con i nomi degli oggetti Amazon Location appropriati.

```
// Amazon Location Service resource names:
const mapName = "explore.map";
const placesName = "explore.place";
const region = "your_region";
const apiKey = "v1.public.a1b2c3d4...

// Initialize a map
async function initializeMap() {
  // Initialize the map
  const mlg1Map = new maplibregl.Map({
    container: "map", // HTML element ID of map element
    center: [-77.03674, 38.891602], // Initial map centerpoint
    zoom: 16, // Initial map zoom
    style: `https://maps.geo.${region}.amazonaws.com/maps/v0/maps/${mapName}/style-
descriptor?key=${apiKey}`, // Defines the appearance of the map and authenticates
using an API key
  });

  // Add navigation control to the top left of the map
  mlg1Map.addControl(new maplibregl.NavigationControl(), "top-left");

  return mlg1Map;
}

async function main() {
  // Create an authentication helper instance using an API key
  const authHelper = await amazonLocationAuthHelper.withAPIKey(apiKey);

  // Initialize map and Amazon Location SDK client
  const map = await initializeMap();
  const client = new amazonLocationClient.LocationClient({
    region,
    ...authHelper.getLocationClientConfig(), // Provides configuration required to
make requests to Amazon Location
  });

  // Variable to hold marker that will be rendered on click
  let marker;

  // On mouse click, display marker and get results:
  map.on("click", async function (e) {
    // Remove any existing marker
    if (marker) {
```



```
        marker.remove();
    }

    // Render a marker on clicked point
    marker = new maplibregl.Marker().setLngLat([e.lngLat.lng,
e.lngLat.lat]).addTo(map);

    // Set up parameters for search call
    let params = {
        IndexName: placesName,
        Position: [e.lngLat.lng, e.lngLat.lat],
        Language: "en",
        MaxResults: "5",
    };

    // Set up command to search for results around clicked point
    const searchCommand = new
amazonLocationClient.SearchPlaceIndexForPositionCommand(params);

    try {
        // Make request to search for results around clicked point
        const data = await client.send(searchCommand);

        // Write JSON response data to HTML
        document.querySelector("#response").textContent = JSON.stringify(data,
undefined, 2);

        // Display place label in an alert box
        alert(data.Results[0].Place.Label);
    } catch (error) {
        // Write JSON response error to HTML
        document.querySelector("#response").textContent = JSON.stringify(error,
undefined, 2);

        // Display error in an alert box
        alert("There was an error searching.");
    }
});
}

main();
```

## Cosa c'è dopo

Hai completato il tutorial di avvio rapido e dovresti avere un'idea di come Amazon Location Service viene utilizzato per creare applicazioni. Per ottenere di più da Amazon Location, puoi consultare le seguenti risorse:

- Approfondisci i [concetti di Amazon Location Service](#)
- Ottieni ulteriori informazioni su [come utilizzare le caratteristiche e le funzionalità di Amazon Location](#)
- Scopri come ampliare questo esempio e creare applicazioni più complesse esaminando [esempi di codice con Amazon Location](#)

## Creazione di un'app per Android

In questa sezione creerai un'applicazione Android con una mappa, la possibilità di cercare una posizione e il tracciamento in primo piano. Innanzitutto, creerai le tue risorse Amazon Location, un'identità Amazon Cognito e una chiave API per la tua applicazione.

### Argomenti

- [Creazione di risorse Amazon Location per la tua app](#)
- [Configurazione dell'autenticazione per l'applicazione](#)
- [Creazione dell'applicazione Android di base](#)
- [Aggiungere una mappa interattiva all'applicazione](#)
- [Aggiungere la ricerca con geocodifica inversa all'applicazione](#)
- [Aggiungere il tracciamento alla tua applicazione](#)
- [Cosa c'è dopo](#)


## Creazione di risorse Amazon Location per la tua app

Se non li possiedi già, devi creare le risorse Amazon Location che l'applicazione utilizzerà. Qui puoi creare una risorsa cartografica per visualizzare le mappe nell'applicazione, un indice dei luoghi per cercare posizioni sulla mappa e un tracker per tracciare un oggetto sulla mappa.

Per aggiungere risorse di localizzazione all'applicazione


1. Scegli lo stile di mappa che desideri utilizzare.

- a. Nella console Amazon Location, nella pagina [Mappe](#), scegli Crea mappa per visualizzare in anteprima gli stili delle mappe.
- b. Aggiungi un nome e una descrizione per la nuova risorsa cartografica. Prendi nota del nome che usi per la risorsa della mappa. Ne avrai bisogno quando creerai il tuo file di script più avanti nel tutorial.
- c. Ti consigliamo di scegliere lo stile di mappa HERE per la tua mappa.

 Note

La scelta di uno stile di mappa consente anche di scegliere il fornitore di dati cartografici da utilizzare. Se la tua applicazione monitora o indirizza le risorse che utilizzi nella tua attività, come veicoli per le consegne o dipendenti, puoi utilizzare HERE solo come fornitore di geolocalizzazione. [Per ulteriori informazioni, consulta la sezione 82 dei termini di servizio.AWS](#)

- d. Accetta i Termini e condizioni di Amazon Location, quindi scegli Crea mappa. Puoi interagire con la mappa che hai scelto: ingrandisci, rimpicciolisci o fai una panoramica in qualsiasi direzione.
  - e. Prendi nota dell'Amazon Resource Name (ARN) visualizzato per la tua nuova risorsa cartografica. Lo utilizzerai per creare l'autenticazione corretta più avanti in questo tutorial.
2. Scegli l'indice dei luoghi che desideri utilizzare.
- a. Nella console Amazon Location, nella pagina degli [indici dei luoghi](#), scegli Crea indice dei luoghi.
  - b. Aggiungi un nome e una descrizione per la nuova risorsa relativa all'indice dei luoghi. Prendi nota del nome che usi per la risorsa dell'indice dei luoghi. Ne avrai bisogno quando creerai il tuo file di script più avanti nel tutorial.
  - c. Scegli un fornitore di dati.

 Note

Nella maggior parte dei casi, scegli il fornitore di dati che corrisponde al fornitore di mappe che hai già scelto. Questo aiuta a garantire che le ricerche corrispondano alle mappe.

Se la tua applicazione monitora o indirizza le risorse che utilizzi nella tua attività, come veicoli per le consegne o dipendenti, puoi utilizzare HERE solo come provider

di geolocalizzazione. [Per ulteriori informazioni, consulta la sezione 82 dei termini di servizio.AWS](#)

- d. Scegli l'opzione di archiviazione dei dati. Per questo tutorial, i risultati non vengono memorizzati, quindi puoi scegliere No, solo monouso.
  - e. Accetta i Termini e condizioni di Amazon Location, quindi scegli Crea indice dei luoghi.
  - f. Prendi nota dell'ARN mostrato per la tua nuova risorsa dell'indice dei luoghi. Lo userai per creare l'autenticazione corretta nella prossima sezione di questo tutorial.
3. Per creare un tracker utilizzando la console Amazon Location.
- a. Apri la [console Amazon Location Service](#).
  - b. Nel riquadro di navigazione a sinistra, scegli Trackers.
  - c. Scegli Crea tracker.
  - d. Compila tutti i campi obbligatori.
  - e. In Filtraggio di posizione, ti consigliamo di utilizzare l'impostazione predefinita: TimeBased
  - f. Scegli Crea tracker per terminare.

## Configurazione dell'autenticazione per l'applicazione

L'applicazione creata in questo tutorial ha un utilizzo anonimo, il che significa che gli utenti non devono accedere AWS per utilizzare l'applicazione. Tuttavia, le API di Amazon Location Service richiedono l'autenticazione per essere utilizzate. Puoi utilizzare le chiavi API o Amazon Cognito per fornire l'autenticazione e l'autorizzazione agli utenti anonimi. Questo tutorial utilizzerà Amazon Cognito e le chiavi API per autenticare la tua applicazione.

### Note

Per ulteriori informazioni sull'utilizzo di Amazon Cognito o delle chiavi API con Amazon Location Service, consulta. [Concessione dell'accesso ad Amazon Location Service](#)

I seguenti tutorial mostrano come configurare l'autenticazione per la mappa, l'indice dei luoghi e il tracker in cui hai creato, oltre a configurare le autorizzazioni per Amazon Location.

## Configura l'autenticazione

1. Accedi alla [console Amazon Location](#) e seleziona le chiavi API dal menu a sinistra.
2. Fai clic su «Crea chiave API». Ricorda che la chiave API deve trovarsi nello stesso AWS account e nella stessa regione delle risorse Amazon Location Service create in precedenza.
3. Inserisci i dettagli richiesti nella pagina «Crea chiave API»:
  - Nome: fornisci un nome per la tua chiave API, ad esempio. MyAppKey
  - Risorse: scegli le risorse Amazon Location Service Map e Place index create in precedenza. Puoi aggiungere più risorse selezionando «Aggiungi risorsa». Ciò consente di utilizzare la chiave API con risorse specifiche.
  - Azioni: specifica le azioni autorizzate per questa chiave API. Come minimo, seleziona `geo:GetMap` e assicurati `geo:SearchPlaceIndexForPosition` che il tutorial funzioni come previsto.
  - Facoltativamente, puoi aggiungere una descrizione, una data di scadenza, dei tag o un riferimento, `https://www.example.com` ad esempio per limitare l'utilizzo della chiave a un dominio specifico, facendo sì che il tutorial funzioni solo all'interno di quel dominio.
4. Fai clic su Crea chiave API per generare la chiave API.
5. Seleziona Mostra chiave API e copia il valore della chiave, ad esempio `v1.public.a1b2c3d4` per utilizzarlo successivamente nel tutorial.

## Crea una policy IAM per il tracciamento

1. Accedi alla console IAM all'indirizzo `https://console.aws.amazon.com/iam/` come utente con autorizzazioni da amministratore.
2. Nel pannello di navigazione, seleziona Policies (Policy).
3. Nel riquadro del contenuto seleziona Create policy (Crea policy).
4. Scegli l'opzione JSON, quindi copia e incolla questa policy JSON nella casella di testo JSON.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "geo:GetMapTile",
```

```

        "geo:GetMapStyleDescriptor",
        "geo:GetMapSprites",
        "geo:GetMapGlyphs",
        "geo:SearchPlaceIndexForPosition",
        "geo:GetDevicePositionHistory",
        "geo:BatchUpdateDevicePosition"
    ],
    "Resource": [
        "arn:aws:geo:{Region}:{Account}:map/{MapName}",
        "arn:aws:geo:{Region}:{Account}:place-index/{IndexName}",
        "arn:aws:geo:{Region}:{Account}:tracker/{TrackerName}"
    ]
}

```

Questo è un esempio di policy per il tracciamento. Per utilizzare l'esempio per la tua politica, sostituisci i TrackerNamesegnaposto RegionAccount, e.

#### Note

Sebbene i pool di identità non autenticati siano destinati all'esposizione su siti Internet non protetti, tieni presente che verranno scambiati con credenziali AWS standard e limitate nel tempo.


È importante definire in modo appropriato i ruoli IAM associati ai pool di identità non autenticati. Per ulteriori informazioni sull'utilizzo e sulla corretta definizione delle politiche in Amazon Cognito con Amazon Location Service, [consulta Concessione dell'accesso ad Amazon Location Service](#).

5. Nella pagina Rivedi e crea, fornisci un nome per il campo del nome della politica. Controlla le autorizzazioni concesse dalla tua politica, quindi scegli Crea politica per salvare il lavoro.

La nuova policy appare nell'elenco delle policy gestite ed è pronta a collegare.

Imposta l'autenticazione per il tracciamento

1. Configura l'autenticazione per la tua applicazione cartografica nella console [Amazon Cognito](#).
2. Apri la pagina dei pool di identità.

 Note

Il pool che crei deve trovarsi nello stesso AWS account e AWS nella stessa regione delle risorse di Amazon Location Service che hai creato nella sezione precedente.

3. Scegli Crea pool di identità.
4. A partire dal passaggio Configure identity pool trust. Per l'autenticazione dell'accesso utente, seleziona Accesso ospite e premi Avanti.
5. Nella pagina Configura le autorizzazioni seleziona Usa un ruolo IAM esistente e inserisci il nome del ruolo IAM che hai creato nel passaggio precedente. Quando sei pronto, premi Avanti per passare alla fase successiva.
6. Nella pagina Configura proprietà, fornisci un nome per il tuo pool di identità. Quindi premi Avanti.
7. Nella pagina Rivedi e crea, esamina tutte le informazioni presenti, quindi premi Crea pool di identità.
8. Apri la pagina dei pool di identità e seleziona il pool di identità che hai appena creato. Quindi copia o annota nello script del browser IdentityPoolId quello che utilizzerai in seguito.

## Creazione dell'applicazione Android di base

In questo tutorial, creerai un'applicazione Android che incorpora una mappa e consente all'utente di trovare cosa si trova in una posizione sulla mappa.

Innanzitutto, crea un'applicazione Kotlin vuota utilizzando la nuova procedura guidata di progetto di Android Studio.

Per creare un'applicazione vuota () AndroidStudio

1. Inizia AndroidStudio. Apri il menu e scegli File, Nuovo, Nuovo progetto.
2. Dalla scheda Telefono e tablet, seleziona Attività vuota, quindi scegli Avanti.
3. Scegli un nome, un nome del pacchetto e una posizione di salvataggio per la tua applicazione.
4. Nell'elenco a discesa per Lingua, seleziona Kotlin.
5. Scegli Fine per creare la tua applicazione vuota.

## Aggiungere una mappa interattiva all'applicazione

Ora che hai creato un'applicazione di base, puoi aggiungere il controllo della mappa all'applicazione. Questo tutorial utilizza le chiavi API per gestire la visualizzazione della mappa. Il controllo della mappa stesso fa parte della [libreria MapLibre nativa](#), con la chiave API e MapLibre i dati della mappa provengono da Amazon Location.

Per aggiungere una mappa alla tua applicazione dovrai eseguire le seguenti azioni:

- Aggiungi la MapLibre dipendenza al tuo progetto.
- Imposta il codice di visualizzazione della mappa con `compose`.
- Scrivi il codice per mostrare la mappa.

Usa la seguente procedura per aggiungere la mappa all'app:

1. Aggiungi la MapLibre dipendenza al tuo progetto
  - a. In AndroidStudio, seleziona il menu Visualizza e scegli Finestre degli strumenti, Progetto. Si aprirà la finestra Progetto, che consente di accedere a tutti i file del progetto.
  - b. Nella finestra del progetto, apri `gradle`, quindi apri il `libs.versions.toml` file nella vista ad albero. Questo aprirà il `libs.versions.toml` file per la modifica. Ora aggiungi la versione e i dati delle librerie seguenti nel `libs.versions.toml` file.

```
[versions]
...
auth = "0.2.4"
tracking = "0.2.4"

[libraries]
...
auth = { group = "software.amazon.location", name = "auth", version.ref =
"auth" }
tracking = { module = "software.amazon.location:tracking", version.ref =
"tracking" }

[plugins]
...
```

- c. Dopo aver finito di modificare il `libs.versions.toml` file, è AndroidStudio necessario risincronizzare il progetto. Nella parte superiore della finestra di `libs.versions.toml`



modifica, AndroidStudio ti chiede di eseguire la sincronizzazione. Seleziona «Sincronizza ora» per sincronizzare il progetto prima di continuare.

- d. Nella finestra del progetto, apri Gradle Scripts nella vista ad albero e seleziona il `build.gradle` file per il tuo modulo applicativo. Questo aprirà il `build.gradle` file per la modifica.
- e. Nella parte inferiore del file, nella sezione delle dipendenze, aggiungi la seguente dipendenza.

```
dependencies {  
    ...  
    implementation(libs.org.maplibre.gl)  
}
```

- f. Dopo aver finito di aggiungere le dipendenze Gradle, Android Studio deve risincronizzare il progetto. Nella parte superiore della finestra di modifica di `build.gradle`, Android Studio, seleziona Sincronizza ora per sincronizzare il progetto prima di continuare.
2. Ora configurerai il codice di visualizzazione della mappa con compose. Utilizza le fasi seguenti:
- a. Dalla finestra del progetto, apri App, Java, *il nome del pacchetto* nella visualizzazione ad albero e vai alla cartella `ui`, all'interno della cartella `ui` crea una directory di visualizzazione.
  - b. All'interno della directory di visualizzazione crea un `MapLoadScreen.kt` file.
  - c. Aggiungi il seguente codice al tuo `MapLoadScreen.kt` file.

```
import androidx.compose.foundation.layout.Box  
import androidx.compose.foundation.layout.fillMaxHeight  
import androidx.compose.foundation.layout.fillMaxWidth  
import androidx.compose.runtime.Composable  
import androidx.compose.ui.Modifier  
import androidx.compose.ui.viewinterop.AndroidView  
import org.maplibre.android.maps.OnMapReadyCallback  
  
@Composable  
fun MapLoadScreen(  
    mapReadyCallback: OnMapReadyCallback,  
) {  
    Box(  
        modifier = Modifier  
            .fillMaxWidth()  
    )  
}
```

```

        .fillMaxHeight(),
    ) {
        MapView(mapReadyCallback)
    }
}

@Composable
fun MapView(mapReadyCallback: OnMapReadyCallback) {
    AndroidView(
        factory = { context ->
            val mapView = org.maplibre.android.maps.MapView(context)
            mapView.onCreate(null)
            mapView.getMapAsync(mapReadyCallback)
            mapView
        },
    )
}

```

### 3. Scrivi il codice per mostrare la mappa.

#### a. Aggiungi il codice seguente al tuo MainActivity.kt file.

```

// ...other imports
import org.maplibre.android.MapLibre
import org.maplibre.android.camera.CameraPosition
import org.maplibre.android.geometry.LatLng
import org.maplibre.android.maps.MapLibreMap
import org.maplibre.android.maps.OnMapReadyCallback
import org.maplibre.android.maps.Style

class MainActivity : ComponentActivity(), OnMapReadyCallback {
    private val region = "YOUR_AWS_REGION"
    private val mapName = "YOUR_AWS_MAP_NAME"
    private val apiKey = "YOUR_AWS_API_KEY"
    override fun onCreate(savedInstanceState: Bundle?) {
        MapLibre.getInstance(this)
        super.onCreate(savedInstanceState)
        setContent {
            TestMapAppTheme {
                Surface(
                    modifier = Modifier.fillMaxSize(),
                    color = MaterialTheme.colorScheme.background
                ) {
                    MapLoadScreen(this)
                }
            }
        }
    }
}

```

```
        }
    }
}

override fun onMapReady(map: MapLibreMap) {
    map.setStyle(
        Style.Builder()
            .fromUri(
                "https://maps.geo.$region.amazonaws.com/maps/v0/maps/
$mapName/style-descriptor?key=$apiKey"
            ),
        ) {
        map.uiSettings.isAttributionEnabled = true
        map.uiSettings.isLogoEnabled = false
        map.uiSettings.attributionGravity = Gravity.BOTTOM or Gravity.END
        val initialPosition = LatLng(47.6160281982247,
-122.32642111977668)
        map.cameraPosition = CameraPosition.Builder()
            .target(initialPosition)
            .zoom(14.0)
            .build()
        }
    }
}
```

- b. Salvare il file `MainActivity.kt`. Ora puoi creare l'applicazione. Per eseguirla, potresti dover configurare un dispositivo in AndroidStudio cui emularla o utilizzarla o utilizzare l'app sul tuo dispositivo. Usa questa app per vedere come si comporta il controllo della mappa. Puoi fare una panoramica trascinandola sulla mappa e pizzicandola per ingrandirla.

Nella sezione successiva, aggiungerai un indicatore sulla mappa e mostrerai l'indirizzo della posizione in cui si trova l'indicatore mentre sposti la mappa.

## Aggiungere la ricerca con geocodifica inversa all'applicazione

Ora aggiungerai la ricerca con geocodifica inversa all'applicazione, in cui troverai gli elementi in una posizione. Per semplificare l'utilizzo di un'app per Android, eseguiremo una ricerca al centro dello schermo. Per trovare una nuova posizione, sposta la mappa nel punto in cui desideri effettuare la ricerca. Posizioneremo un indicatore al centro della mappa per mostrare dove stiamo cercando.

L'aggiunta di una ricerca con geocodifica inversa sarà composta da due parti.

- Aggiungi un marker al centro dello schermo per mostrare all'utente dove stiamo cercando.
- Aggiungi una casella di testo per i risultati, quindi cerca cosa si trova nella posizione del marker e mostralo nella casella di testo.

Per aggiungere un marker alla tua applicazione

1. Salva questa immagine nel tuo progetto nella app/res/drawable cartella come `red_marker.png` (puoi anche accedere all'immagine da [GitHub](#)). In alternativa, puoi creare la tua immagine. Puoi anche usare un file.png con trasparenza per le parti che non vuoi mostrare.
2. Aggiungi il codice seguente al tuo `MapLoadScreenfile.kt`.

```
// ...other imports
import androidx.compose.foundation.Image
import androidx.compose.foundation.layout.size
import androidx.compose.ui.Alignment
import androidx.compose.ui.res.painterResource
import androidx.compose.ui.unit.dp
import com.amazon.testmapapp.R

@Composable
fun MapLoadScreen(
    mapReadyCallback: OnMapReadyCallback,
) {
    Box(
        modifier = Modifier
            .fillMaxWidth()
            .fillMaxHeight(),
    ) {
        MapView(mapReadyCallback)
        Box(
            modifier = Modifier
                .align(Alignment.Center),
        ) {
            Image(
                painter = painterResource(id = R.drawable.red_marker),
                contentDescription = "marker",
                modifier = Modifier
                    .size(40.dp)
                    .align(Alignment.Center),
            )
        }
    }
}
```

```
    }  
  }  
}  
  
@Composable  
fun MapView(mapReadyCallback: OnMapReadyCallback) {  
    AndroidView(  
        factory = { context ->  
            val mapView = org.maplibre.android.maps.MapView(context)  
            mapView.onCreate(null)  
            mapView.getMapAsync(mapReadyCallback)  
            mapView  
        },  
    )  
}
```

3. Crea ed esegui la tua app per visualizzare in anteprima la funzionalità.

La tua app ora ha un indicatore sullo schermo. In questo caso, si tratta di un'immagine statica che non si muove. Viene utilizzato per mostrare il centro della visualizzazione della mappa, dove cercheremo. Nella procedura seguente, aggiungeremo la ricerca in quella posizione.

Per aggiungere alla tua app la ricerca con geocodifica inversa in una località

1. Nella finestra del progetto, apri gradle in un `libs.versions.toml` file nella vista ad albero. Questo aprirà il `libs.versions.toml` file per la modifica. Ora aggiungi la versione e i dati delle librerie seguenti nel `libs.versions.toml` file.

```
[versions]  
...  
okhttp = "4.12.0"  
  
[libraries]  
...  
com-squareup-okhttp3 = { group = "com.squareup.okhttp3", name = "okhttp",  
    version.ref = "okhttp" }  
  
[plugins]  
...
```

2. Dopo aver finito di modificare il `libs.versions.toml` file, è AndroidStudio necessario risincronizzare il progetto. Nella parte superiore della finestra di `libs.versions.toml`

modifica, AndroidStudio ti chiede di eseguire la sincronizzazione. Seleziona «Sincronizza ora» per sincronizzare il progetto prima di continuare.

3. Nella finestra del progetto, apri Gradle Scripts nella vista ad albero e seleziona il `build.gradle` file per il tuo modulo applicativo. Questo aprirà il `build.gradle` file per la modifica.
4. Nella parte inferiore del file, nella sezione delle dipendenze, aggiungi la seguente dipendenza.

```
dependencies {  
    ...  
    implementation(libs.com.squareup.okhttp3)  
}
```

5. Dopo aver finito di modificare le dipendenze di Gradle, è AndroidStudio necessario risincronizzare il progetto. Nella parte superiore della finestra di `build.gradle` modifica, ti AndroidStudio chiede di eseguire la sincronizzazione. Seleziona questa opzione SyncNow per sincronizzare il progetto prima di continuare.
6. Ora nella visualizzazione ad albero aggiungi i dati alla directory delle richieste e crea la classe di `ReverseGeocodeRequest.kt` dati. Aggiungi il seguente codice alla classe.

```
import com.google.gson.annotations.SerializedName  
  
data class ReverseGeocodeRequest(  
    @SerializedName("Language")  
    val language: String,  
    @SerializedName("MaxResults")  
    val maxResults: Int,  
    @SerializedName("Position")  
    val position: List<Double>  
)
```

7. Ora nella visualizzazione ad albero aggiungi i dati alla directory delle risposte e crea la classe di `ReverseGeocodeResponse.kt` dati. Aggiungi il seguente codice al suo interno.

```
import com.google.gson.annotations.SerializedName  
  
data class ReverseGeocodeResponse(  
    @SerializedName("Results")  
    val results: List<Result>  
)  
  
data class Result(  
    ...  
)
```

```
        @SerializedName("Place")
        val place: Place
    )

    data class Place(
        @SerializedName("Label")
        val label: String
    )
```

8. Ora, dalla finestra del progetto, apri App, Java, *il nome del pacchetto* nella visualizzazione ad albero e vai alla cartella ui, all'interno della cartella ui create la directory ViewModel.
9. All'interno della directory ViewModel crea MainViewModel.kt il file.
10. Aggiungi il seguente codice al tuo MainViewModel.kt file.

```
import androidx.compose.runtime.getValue
import androidx.compose.runtime.mutableStateOf
import androidx.compose.runtime.setValue
import androidx.lifecycle.ViewModel
import com.amazon.testmapapp.data.request.ReverseGeocodeRequest
import com.amazon.testmapapp.data.response.ReverseGeocodeResponse
import com.google.gson.Gson
import java.io.IOException
import okhttp3.Call
import okhttp3.Callback
import okhttp3.MediaType.Companion.toMediaTypeOrNull
import okhttp3.OkHttpClient
import okhttp3.Request
import okhttp3.RequestBody.Companion.toRequestBody
import okhttp3.Response
import org.maplibre.android.geometry.LatLng
import org.maplibre.android.maps.MapLibreMap

class MainViewModel : ViewModel() {
    var label by mutableStateOf("")
    var isLabelAdded: Boolean by mutableStateOf(false)
    var client = OkHttpClient()
    var mapLibreMap: MapLibreMap? = null

    fun reverseGeocode(latLng: LatLng, apiKey: String) {
        val region = "YOUR_AWS_REGION"
        val indexName = "YOUR_AWS_PLACE_INDEX"
        val url =
```

```
        "https://places.geo.${region}.amazonaws.com/places/v0/indexes/
        ${indexName}/search/position?key=${apiKey}"

        val requestBody = ReverseGeocodeRequest(
            language = "en",
            maxResults = 1,
            position = listOf(latLng.longitude, latLng.latitude)
        )
        val json = Gson().toJson(requestBody)

        val mediaType = "application/json".toMediaTypeOrNull()
        val request = Request.Builder()
            .url(url)
            .post(json.toRequestBody(mediaType))
            .build()

        client.newCall(request).enqueue(object : Callback {
            override fun onFailure(call: Call, e: IOException) {
                e.printStackTrace()
            }

            override fun onResponse(call: Call, response: Response) {
                if (response.isSuccessful) {
                    val jsonResponse = response.body?.string()

                    val reverseGeocodeResponse =
                        Gson().fromJson(jsonResponse,
ReverseGeocodeResponse::class.java)

                    val responseLabel =
reverseGeocodeResponse.results.firstOrNull()?.place?.label

                    if (responseLabel != null) {
                        label = responseLabel
                        isLabelAdded = true
                    }
                }
            }
        })
    }
}
```



11. Se non è già aperto, apri il `MapLoadScreen.kt` file, come nella procedura precedente. Aggiungere il codice seguente. Verrà creata una vista di composizione del testo in cui verranno visualizzati i risultati della ricerca con geocodifica inversa nella posizione.

```
// ...other imports
import androidx.compose.foundation.background
import androidx.compose.foundation.layout.Arrangement
import androidx.compose.foundation.layout.Column
import androidx.compose.foundation.layout.Spacer
import androidx.compose.foundation.layout.fillMaxSize
import androidx.compose.foundation.layout.height
import androidx.compose.foundation.layout.padding
import androidx.compose.material3.Text
import androidx.compose.ui.graphics.Color
import androidx.compose.ui.platform.testTag
import androidx.compose.ui.semantics.contentDescription
import androidx.compose.ui.semantics.semantics
import androidx.compose.ui.unit.sp
import com.amazon.testmapapp.ui.viewModel.MainViewModel

@Composable
fun MapLoadScreen(
    mapReadyCallback: OnMapReadyCallback,
    mainViewModel: MainViewModel,
) {
    Box(
        modifier = Modifier
            .fillMaxWidth()
            .fillMaxHeight(),
    ) {
        MapView(mapReadyCallback)
        Box(
            modifier = Modifier
                .align(Alignment.Center),
        ) {
            Image(
                painter = painterResource(id = R.drawable.red_marker),
                contentDescription = "marker",
                modifier = Modifier
                    .size(40.dp)
                    .align(Alignment.Center),
            )
        }
    }
}
```

```
    }
    if (mainViewModel.isLabelAdded) {
        Column(
            modifier = Modifier.fillMaxSize(),
            verticalArrangement = Arrangement.Bottom
        ) {
            Box(
                modifier = Modifier
                    .fillMaxWidth()
                    .background(Color.White),
            ) {
                Text(
                    text = mainViewModel.label,
                    modifier = Modifier
                        .padding(16.dp)
                        .align(Alignment.Center)
                        .testTag("label")
                        .semantics {
                            contentDescription = "label"
                        },
                    fontSize = 14.sp,
                )
            }
            Spacer(modifier = Modifier.height(80.dp))
        }
    }
}

@Composable
fun MapView(mapReadyCallback: OnMapReadyCallback) {
    AndroidView(
        factory = { context ->
            val mapView = org.maplibre.android.maps.MapView(context)
            mapView.onCreate(null)
            mapView.getMapAsync(mapReadyCallback)
            mapView
        },
    )
}
```

12. Nell'app, in java, nella cartella del nome del pacchetto in AndroidStudio, apri il file. MainActivity.kt Modifica il codice come mostrato.

```
// ...other imports
import androidx.activity.viewModels
import com.amazon.testmapapp.ui.viewModel.MainViewModel

class MainActivity : ComponentActivity(), OnMapReadyCallback,
MapLibreMap.OnCameraMoveStartedListener, MapLibreMap.OnCameraIdleListener {

    private val mainViewModel: MainViewModel by viewModels()
    private val region = "YOUR_AWS_REGION"
    private val mapName = "YOUR_AWS_MAP_NAME"
    private val apiKey = "YOUR_AWS_API_KEY"
    override fun onCreate(savedInstanceState: Bundle?) {
        MapLibre.getInstance(this)
        super.onCreate(savedInstanceState)
        setContent {
            TestMapAppTheme {
                Surface(
                    modifier = Modifier.fillMaxSize(),
                    color = MaterialTheme.colorScheme.background
                ) {
                    MapLoadScreen(this, mainViewModel)
                }
            }
        }
    }

    override fun onMapReady(map: MapLibreMap) {
        map.setStyle(
            Style.Builder()
                .fromUri(
                    "https://maps.geo.$region.amazonaws.com/maps/v0/maps/$mapName/
style-descriptor?key=$apiKey"
                ),
        ) {
            map.uiSettings.isAttributionEnabled = true
            map.uiSettings.isLogoEnabled = false
            map.uiSettings.attributionGravity = Gravity.BOTTOM or Gravity.END
            val initialPosition = LatLng(47.6160281982247, -122.32642111977668)
            map.cameraPosition = CameraPosition.Builder()
                .target(initialPosition)
                .zoom(14.0)
                .build()
        }
    }
}
```

```

        map.addOnCameraMoveStartedListener(this)
        map.addOnCameraIdleListener(this)
        map.cameraPosition.target?.let { latLng ->
            mainViewModel.reverseGeocode(
                LatLng(
                    latLng.latitude,
                    latLng.longitude
                ), apiKey
            )
        }
    }
}

override fun onCameraMoveStarted(p0: Int) {
    mainViewModel.label = ""
    mainViewModel.isLabelAdded = false
}

override fun onCameraIdle() {
    if (!mainViewModel.isLabelAdded) {
        mainViewModel.mapLibreMap?.cameraPosition?.target?.let { latLng ->
            mainViewModel.reverseGeocode(
                LatLng(
                    latLng.latitude,
                    latLng.longitude
                ), apiKey
            )
        }
    }
}
}
}

```

Questo codice funziona con la visualizzazione della mappa. La posizione della telecamera virtuale definisce la visualizzazione della mappa in MapLibre. Lo spostamento della mappa può essere considerato come lo spostamento di quella telecamera virtuale.

- ViewModel ha una variabile label: questa variabile imposta i dati nella vista di composizione del testo.
- onMapReady ha una variabile label: Questa variabile imposta i dati nella vista di composizione del testo. ----sep----:Questa funzione viene aggiornata per registrare due nuovi eventi.

- L'onCameraMoveevento si verifica ogni volta che l'utente sposta la mappa. In generale, quando spostiamo la mappa, vogliamo nascondere la ricerca finché l'utente non ha finito di spostare la mappa.
- L'onCameraIdleevento si verifica quando l'utente sospende lo spostamento della mappa. Questo evento richiama la nostra funzione di geocodifica inversa per effettuare ricerche al centro della mappa.
- `reverseGeocode(latLng: LatLng, apiKey: String)`: Questa funzione, chiamata nell'evento `onCameraIdle`, cerca una posizione al centro della mappa e aggiorna l'etichetta per mostrare i risultati. Utilizza il target della fotocamera, che definisce il centro della mappa (dove la fotocamera sta guardando).

13. Salva i tuoi file e crea ed esegui l'app per vedere se funziona.

La tua applicazione di avvio rapido con funzionalità di ricerca è completa.

## Aggiungere il tracciamento alla tua applicazione

Per aggiungere il tracciamento alla tua applicazione di esempio, segui questi passaggi:

1. Aggiungi le dipendenze dell'SDK di tracciamento e autenticazione al tuo progetto.
  2. Includi le voci di autorizzazione e servizio nel tuo file.xml AndroidManifest.
  3. Imposta il codice del pulsante di avvio/arresto del tracciamento con `compose`.
  4. Aggiungi codice per creare un `LocationTracker` oggetto e avviare e interrompere il tracciamento.
  5. Crea un percorso di prova con Android Emulator.
1. Aggiungi le dipendenze dell'SDK di tracciamento e autenticazione al tuo progetto.
    - a. Nella finestra del progetto, apri `gradle`, quindi apri il `libs.versions.toml` file nella vista ad albero. Questo aprirà il `libs.versions.toml` file per la modifica. Ora aggiungi la versione e i dati delle librerie seguenti nel `libs.versions.toml` file.

```
[versions]
...
auth = "0.0.1"
tracking = "0.0.1"

[libraries]
...
```

```
auth = { group = "software.amazon.location", name = "auth", version.ref =
"auth" }
tracking = { module = "software.amazon.location:tracking", version.ref =
"tracking" }

[plugins]
...
```

- b. Dopo aver finito di modificare il `libs.versions.toml` file, è AndroidStudio necessario risincronizzare il progetto. Nella parte superiore della finestra di `libs.versions.toml` modifica, AndroidStudio ti chiede di eseguire la sincronizzazione. Seleziona «Sincronizza ora» per sincronizzare il progetto prima di continuare.
- c. Nella finestra del progetto, apri Gradle Scripts nella vista ad albero e seleziona il `build.gradle` file per il tuo modulo applicativo. Questo aprirà il `build.gradle` file per la modifica.
- d. Nella parte inferiore del file, nella sezione delle dipendenze, aggiungi la seguente dipendenza.

```
dependencies {
    ...
    implementation(libs.auth)
    implementation(libs.tracking)
}
```

- e. Dopo aver finito di modificare le dipendenze di Gradle, è AndroidStudio necessario risincronizzare il progetto. Nella parte superiore della finestra di modifica `build.gradle`, ti chiede di eseguire la sincronizzazione. AndroidStudio Seleziona di SyncNow sincronizzare il progetto prima di continuare.
2. Includi le autorizzazioni e le voci di servizio nel tuo `AndroidManifest.xml` file.
    - Per includere le voci di autorizzazione e servizio corrette nel tuo `fileAndroidManifest.xml` file, aggiorna il file con il codice seguente:

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools">

    <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"/>
    <uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION"/>
    <uses-permission android:name="android.permission.INTERNET"/>
```

```

<application
    android:allowBackup="true"
    android:dataExtractionRules="@xml/data_extraction_rules"
    android:fullBackupContent="@xml/backup_rules"
    android:icon="@mipmap/ic_launcher"
    android:label="@string/app_name"
    android:roundIcon="@mipmap/ic_launcher_round"
    android:supportsRtl="true"
    android:theme="@style/Theme.AndroidQuickStartApp"
    tools:targetApi="31">
    <activity
        android:name=".MainActivity"
        android:exported="true"
        android:label="@string/app_name"
        android:theme="@style/Theme.AndroidQuickStartApp">
        <intent-filter>
            <action android:name="android.intent.action.MAIN" />

            <category android:name="android.intent.category.LAUNCHER" />
        </intent-filter>
    </activity>
</application>

</manifest>

```

3. Imposta il codice del pulsante di avvio/arresto del tracciamento con compose.
  - a. Aggiungi due immagini di Play e Pause in risoluzione nel file drawable denominato ic\_pause e ic\_play. Puoi anche accedere all'immagine da [GitHub](#)
  - b. Se non è già aperto, apri il MapLoadScreen.kt file, come nella procedura precedente. Aggiungere il codice seguente. Questo creerà una visualizzazione del pulsante di composizione in cui potremo fare clic su di esso per avviare e interrompere il tracciamento.

```

// ...other imports
import androidx.compose.material3.Button
import androidx.compose.material3.ButtonDefaults

@Composable
fun MapLoadScreen(
    mapReadyCallback: OnMapReadyCallback,
    mainViewModel: MainViewModel,
    onStartStopTrackingClick: () -> Unit
) {

```

```
Box(
    modifier = Modifier
        .fillMaxWidth()
        .fillMaxHeight(),
) {
    MapView(mapReadyCallback)
    Box(
        modifier = Modifier
            .align(Alignment.Center),
    ) {
        Image(
            painter = painterResource(id = R.drawable.red_marker),
            contentDescription = "marker",
            modifier = Modifier
                .size(40.dp)
                .align(Alignment.Center),
        )
    }
    if (mainViewModel.isLabelAdded) {
        Column(
            modifier = Modifier.fillMaxSize(),
            verticalArrangement = Arrangement.Bottom
        ) {
            Box(
                modifier = Modifier
                    .fillMaxWidth()
                    .background(Color.White),
            ) {
                Text(
                    text = mainViewModel.label,
                    modifier = Modifier
                        .padding(16.dp)
                        .align(Alignment.Center)
                        .testTag("label")
                        .semantics {
                            contentDescription = "label"
                        },
                    fontSize = 14.sp,
                )
            }
            Spacer(modifier = Modifier.height(80.dp))
        }
    }
}
Column(
```



```

        modifier = Modifier
            .fillMaxSize()
            .padding(bottom = 16.dp),
        horizontalAlignment = Alignment.CenterHorizontally,
        verticalArrangement = Arrangement.Bottom,
    ) {
        Button(
            onClick = onStartStopTrackingClick,
            modifier = Modifier
                .padding(horizontal = 16.dp)
        ) {
            Text(
                text = if
(mainViewModel.isLocationTrackingForegroundActive) "Stop tracking" else "Start
tracking",
                color = Color.Black
            )
            Spacer(modifier = Modifier.size(ButtonDefaults.IconSpacing))
            Image(
                painter = painterResource(id = if
(mainViewModel.isLocationTrackingForegroundActive) R.drawable.ic_pause else
R.drawable.ic_play),
                contentDescription = if
(mainViewModel.isLocationTrackingForegroundActive) "stop_tracking" else
"start_tracking"
            )
        }
    }
}

@Composable
fun MapView(mapReadyCallback: OnMapReadyCallback) {
    AndroidView(
        factory = { context ->
            val mapView = org.maplibre.android.maps.MapView(context)
            mapView.onCreate(null)
            mapView.getMapAsync(mapReadyCallback)
            mapView
        },
    )
}

```

#### 4. Aggiungi codice per creare un LocationTracker oggetto e avviare e interrompere il tracciamento.

- a. Aggiungi il seguente codice all'interno del `MainViewModel.kt` file.

```
...
var isLocationTrackingForegroundActive: Boolean by mutableStateOf(false)
var locationTracker: LocationTracker? = null
```

- b. Aggiungi il codice seguente al tuo `MainActivity.kt` file.

```
// ...other imports
import software.amazon.location.auth.AuthHelper
import software.amazon.location.auth.LocationCredentialsProvider
import software.amazon.location.tracking.LocationTracker
import software.amazon.location.tracking.aws.LocationTrackingCallback
import software.amazon.location.tracking.config.LocationTrackerConfig
import software.amazon.location.tracking.database.LocationEntry
import software.amazon.location.tracking.filters.DistanceLocationFilter
import software.amazon.location.tracking.filters.TimeLocationFilter
import software.amazon.location.tracking.util.TrackingSdkLogLevel

class MainActivity : ComponentActivity(), OnMapReadyCallback,
    MapLibreMap.OnCameraMoveStartedListener, MapLibreMap.OnCameraIdleListener {

    private val mainViewModel: MainViewModel by viewModels()
    private val poolId = "YOUR_AWS_IDENTITY_POOL_ID"
    private val trackerName = "YOUR_AWS_TRACKER_NAME"
    private val region = "YOUR_AWS_REGION"
    private val mapName = "YOUR_AWS_MAP_NAME"
    private val apiKey = "YOUR_AWS_API_KEY"
    private val coroutineScope = MainScope()
    private lateinit var locationCredentialsProvider:
LocationCredentialsProvider
    private lateinit var authHelper: AuthHelper

    override fun onCreate(savedInstanceState: Bundle?) {
        MapLibre.getInstance(this)
        super.onCreate(savedInstanceState)
        setContent {
            TestMapAppTheme {
                Surface(
                    modifier = Modifier.fillMaxSize(),
                    color = MaterialTheme.colorScheme.background
                ) {

```

```

        MapLoadScreen(this, mainViewModel, onStartStopTrackingClick
= {
            if (mainViewModel.isLocationTrackingForegroundActive) {
                mainViewModel.isLocationTrackingForegroundActive =
false
                mainViewModel.locationTracker?.stop()
            } else {
                if (checkLocationPermission(this))
return@MapLoadScreen
                mainViewModel.isLocationTrackingForegroundActive =
true

                mainViewModel.locationTracker?.start(locationTrackingCallback = object :
                    LocationTrackingCallback {
                        override fun
onLocationAvailabilityChanged(locationAvailable: Boolean) {
                            }

                            override fun onLocationReceived(location:
LocationEntry) {
                            }

                            override fun onUploadSkipped(entries:
LocationEntry) {
                            }

                            override fun onUploadStarted(entries:
List<LocationEntry>) {
                            }

                            override fun onUploaded(entries:
List<LocationEntry>) {
                            }
                        })
                    })
            }
        }
    }
    authenticateUser()
}

private fun authenticateUser() {

```

```
coroutineScope.launch {
    authHelper = AuthHelper(applicationContext)
    locationCredentialsProvider =
authHelper.authenticateWithCognitoIdentityPool(
    poolId,
)
    locationCredentialsProvider.let {
        val config = LocationTrackerConfig(
            trackerName = trackerName,
            logLevel = TrackingSdkLogLevel.DEBUG,
            latency = 1000,
            frequency = 5000,
            waitForAccurateLocation = false,
            minUpdateIntervalMillis = 5000,
        )
        mainViewModel.locationTracker = LocationTracker(
            applicationContext,
            it,
            config,
        )
    }

mainViewModel.locationTracker?.enableFilter(TimeLocationFilter())

mainViewModel.locationTracker?.enableFilter(DistanceLocationFilter())
    }
}

private fun checkLocationPermission(context: Context) =
    ActivityCompat.checkSelfPermission(
        context,
        Manifest.permission.ACCESS_FINE_LOCATION,
    ) != PackageManager.PERMISSION_GRANTED &&
    ActivityCompat.checkSelfPermission(
        context,
        Manifest.permission.ACCESS_COARSE_LOCATION,
    ) != PackageManager.PERMISSION_GRANTED

override fun onMapReady(map: MapLibreMap) {
    map.setStyle(
        Style.Builder()
            .fromUri(
                "https://maps.geo.$region.amazonaws.com/maps/v0/maps/
$mapName/style-descriptor?key=$apiKey"
            )
    )
}
```

```
        ),
    ) {
        mainViewModel.mapLibreMap = map
        map.uiSettings.isAttributionEnabled = true
        map.uiSettings.isLogoEnabled = false
        map.uiSettings.attributionGravity = Gravity.BOTTOM or Gravity.END
        val initialPosition = LatLng(47.6160281982247, -122.32642111977668)
        map.cameraPosition = CameraPosition.Builder()
            .target(initialPosition)
            .zoom(14.0)
            .build()

        map.addOnCameraMoveStartedListener(this)
        map.addOnCameraIdleListener(this)
        map.cameraPosition.target?.let { latLng ->
            mainViewModel.reverseGeocode(
                LatLng(
                    latLng.latitude,
                    latLng.longitude
                ), apiKey
            )
        }
    }
}

override fun onCameraMoveStarted(p0: Int) {
    mainViewModel.label = ""
    mainViewModel.isLabelAdded = false
}

override fun onCameraIdle() {
    if (!mainViewModel.isLabelAdded) {
        mainViewModel.mapLibreMap?.cameraPosition?.target?.let { latLng ->
            mainViewModel.reverseGeocode(
                LatLng(
                    latLng.latitude,
                    latLng.longitude
                ), apiKey
            )
        }
    }
}
}
```

Il codice precedente mostra come creare un `LocationTracker` oggetto con `AuthHelper` e come avviare e interrompere il tracciamento con `LocationTracker`.

- `authenticateUser()`: Questo metodo crea `AuthHelper` `LocationTracker` oggetti.
- `onStartStopTrackingClick`: Questo callback viene attivato quando l'utente fa clic sul pulsante `Avvia/interrompi il tracciamento`, che avvia/interrompe il tracciamento con `Tracking SDK`.

#### 5. Crea un percorso di prova con Android Emulator.

- Apri l'emulatore avviando l'AVD utilizzando Android Studio.
- Apri `Extended Controls` facendo clic sull'icona `Altro` (tre punti) nella barra degli strumenti dell'emulatore.
- Apri `Posizione` selezionando `Posizione` dalla barra laterale.
- Crea un percorso con dati `GPX` o facendo clic sulla mappa e scegliendo i dati di origine e destinazione.
- Inizia la simulazione facendo clic su `PLAY ROUTE` per iniziare a simulare il percorso `GPS`.
- Testa l'applicazione eseguendo l'applicazione e osservando come gestisce il percorso simulato.

Questa è la demo completa dell'applicazione `Android Quick Start`.

## Cosa c'è dopo

Il codice sorgente di questa applicazione è disponibile su [GitHub](#).

Per ottenere di più da `Amazon Location`, puoi consultare le seguenti risorse:

- Approfondisci i [concetti di Amazon Location Service](#)
- Ottieni ulteriori informazioni su [come utilizzare le caratteristiche e le funzionalità di Amazon Location](#)
- Scopri come ampliare questo esempio e creare applicazioni più complesse esaminando [esempi di codice con Amazon Location](#).

# Creazione di un'app per iOS

In questa sezione, creerai un'applicazione iOS con la possibilità di cercare in una posizione e di tracciare in primo piano. Innanzitutto, creerai le tue risorse Amazon Location e un'identità Amazon Cognito per la tua applicazione.

## Argomenti


- [Creazione di risorse Amazon Location per la tua app](#)
- [Configurazione dell'autenticazione per l'applicazione](#)
- [Creazione dell'applicazione iOS di base](#)
- [Impostazione del codice iniziale](#)
- [Aggiungere una mappa interattiva all'applicazione](#)
- [Aggiungere la ricerca all'applicazione](#)
- [Aggiungere il tracciamento alla tua applicazione](#)
- [Cosa c'è dopo](#)

## Creazione di risorse Amazon Location per la tua app

Se non li possiedi già, devi creare le risorse Amazon Location che l'applicazione utilizzerà. Creerai una risorsa cartografica per visualizzare le mappe nell'applicazione, un indice dei luoghi per cercare posizioni sulla mappa e un tracker per tracciare un oggetto sulla mappa.


Per aggiungere risorse di localizzazione all'applicazione

1. Scegli lo stile di mappa che desideri utilizzare.
  - a. Nella console Amazon Location, nella pagina [Mappe](#), scegli Crea mappa per visualizzare in anteprima gli stili delle mappe.
  - b. Aggiungi un nome e una descrizione per la nuova risorsa cartografica. Prendi nota del nome che usi per la risorsa della mappa. Ne avrai bisogno quando creerai il tuo file di script più avanti nel tutorial.
  - c. Scegli una mappa.

 Note

La scelta di uno stile di mappa consente anche di scegliere il fornitore di dati cartografici da utilizzare. Se la tua applicazione monitora o indirizza le risorse che utilizzi nella tua attività, come veicoli per le consegne o dipendenti, puoi utilizzare HERE solo come fornitore di geolocalizzazione. [Per ulteriori informazioni, consulta la sezione 82 dei termini di servizio.AWS](#)

- d. Accetta i Termini e condizioni di Amazon Location, quindi scegli Crea mappa. Puoi interagire con la mappa che hai scelto: ingrandisci, rimpicciolisci o fai una panoramica in qualsiasi direzione.
  - e. Prendi nota dell'Amazon Resource Name (ARN) visualizzato per la tua nuova risorsa cartografica. Lo utilizzerai per creare l'autenticazione corretta più avanti in questo tutorial.
2. Scegli l'indice dei luoghi che desideri utilizzare.
- a. Nella console Amazon Location, nella pagina degli [indici dei luoghi](#), scegli Crea indice dei luoghi.
  - b. Aggiungi un nome e una descrizione per la nuova risorsa relativa all'indice dei luoghi. Prendi nota del nome che usi per la risorsa dell'indice dei luoghi. Ne avrai bisogno quando creerai il tuo file di script più avanti nel tutorial.
  - c. Scegli un fornitore di dati.

 Note

Nella maggior parte dei casi, scegli il fornitore di dati che corrisponde al fornitore di mappe che hai già scelto. Questo aiuta a garantire che le ricerche corrispondano alle mappe.

Se la tua applicazione monitora o indirizza le risorse che utilizzi nella tua attività, come veicoli per le consegne o dipendenti, puoi utilizzare HERE solo come provider di geolocalizzazione. [Per ulteriori informazioni, consulta la sezione 82 dei termini di servizio.AWS](#)

- d. Scegli l'opzione di archiviazione dei dati. Per questo tutorial, i risultati non vengono memorizzati, quindi puoi scegliere No, solo monouso.
- e. Accetta i Termini e condizioni di Amazon Location, quindi scegli Crea indice dei luoghi.



- f. Prendi nota dell'ARN mostrato per la tua nuova risorsa dell'indice dei luoghi. Lo userai per creare l'autenticazione corretta nella prossima sezione di questo tutorial.
3. Per creare un tracker utilizzando la console Amazon Location.
    - a. Apri la [console Amazon Location Service](#).
    - b. Nel riquadro di navigazione a sinistra, scegli Trackers.
    - c. Scegli Crea tracker.
    - d. Compila tutti i campi obbligatori.
    - e. In Filtraggio di posizione, ti consigliamo di utilizzare l'impostazione predefinita: TimeBased
    - f. Scegli Crea tracker per terminare.

## Configurazione dell'autenticazione per l'applicazione

L'applicazione creata in questo tutorial ha un utilizzo anonimo, il che significa che gli utenti non devono accedere AWS per utilizzare l'applicazione. Tuttavia, le API di Amazon Location Service richiedono l'autenticazione per essere utilizzate. Utilizzerai Amazon Cognito per fornire l'autenticazione e l'autorizzazione agli utenti anonimi. Questo tutorial utilizzerà Amazon Cognito per autenticare la tua applicazione.

### Note

Per ulteriori informazioni sull'utilizzo di Amazon Cognito con Amazon Location Service, consulta [Concessione dell'accesso ad Amazon Location Service](#)

I seguenti tutorial mostrano come configurare l'autenticazione per la mappa, l'indice dei luoghi e il tracker in cui hai creato, oltre a configurare le autorizzazioni per Amazon Location.

### Crea una policy IAM per il tracciamento

1. Accedi alla console IAM all'indirizzo <https://console.aws.amazon.com/iam/> come utente con autorizzazioni da amministratore.
2. Nel pannello di navigazione, seleziona Policies (Policy).
3. Nel riquadro del contenuto seleziona Create policy (Crea policy).
4. Scegli l'opzione JSON, quindi copia e incolla questa policy JSON nella casella di testo JSON.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "geo:GetMapTile",
        "geo:GetMapStyleDescriptor",
        "geo:GetMapSprites",
        "geo:GetMapGlyphs",
        "geo:SearchPlaceIndexForPosition",
        "geo:GetDevicePositionHistory",
        "geo:BatchUpdateDevicePosition"
      ],
      "Resource": [
        "arn:aws:geo:{Region}:{Account}:map/{MapName}",
        "arn:aws:geo:{Region}:{Account}:place-index/{IndexName}",
        "arn:aws:geo:{Region}:{Account}:tracker/{TrackerName}"
      ]
    }
  ]
}
```

Questo è un esempio di policy per il tracciamento. Per utilizzare l'esempio per la tua politica, sostituisci i TrackerName, Region, Account, IndexName, MapName e.

#### Note

Sebbene i pool di identità non autenticati siano destinati all'esposizione su siti Internet non protetti, tieni presente che verranno scambiati con credenziali standard a tempo limitato. AWS

È importante definire in modo appropriato i ruoli IAM associati ai pool di identità non autenticati. Per ulteriori informazioni sull'utilizzo e sulla corretta definizione delle politiche in Amazon Cognito con Amazon Location Service, [consulta Concessione dell'accesso ad Amazon Location Service](#).

5. Nella pagina Rivedi e crea, fornisci un nome per il campo del nome della politica. Controlla le autorizzazioni concesse dalla tua politica, quindi scegli Crea politica per salvare il lavoro.

La nuova policy appare nell'elenco delle policy gestite ed è pronta a collegare.

Imposta l'autenticazione per il tracciamento

1. Configura l'autenticazione per la tua applicazione cartografica nella console [Amazon Cognito](#).
2. Apri la pagina dei pool di identità.

#### Note

Il pool che crei deve trovarsi nello stesso AWS account e AWS nella stessa regione delle risorse di Amazon Location Service che hai creato nella sezione precedente.

3. Scegli Crea pool di identità.
4. A partire dal passaggio Configure identity pool trust. Per l'autenticazione dell'accesso utente, seleziona Accesso ospite e premi Avanti.
5. Nella pagina Configura le autorizzazioni seleziona Usa un ruolo IAM esistente e inserisci il nome del ruolo IAM che hai creato nel passaggio precedente. Quando sei pronto, premi Avanti per passare alla fase successiva.
6. Nella pagina Configura proprietà, fornisci un nome per il tuo pool di identità. Quindi premi Avanti.
7. Nella pagina Rivedi e crea, esamina tutte le informazioni presenti, quindi premi Crea pool di identità.
8. Apri la pagina dei pool di identità e seleziona il pool di identità che hai appena creato. Quindi copia o annota nello script del browser IdentityPoolId quello che utilizzerai in seguito.

## Creazione dell'applicazione iOS di base

In questo tutorial, creerai un'applicazione iOS che incorpora una mappa e consente all'utente di trovare cosa si trova in una posizione sulla mappa.

Per prima cosa, creiamo un'applicazione Swift usando la procedura guidata di progetto di Xcode.

Per creare un'applicazione vuota (Xcode)

1. Apri Xcode e dal menu scegli File, Nuovo, Nuovo progetto.
2. Dalla scheda iOS, seleziona App, quindi scegli Avanti.
3. Fornisci un nome di prodotto, un identificatore dell'organizzazione e SwiftUI inseriscilo nel campo Interfaccia. Scegli Avanti per finalizzare la selezione.

4. Seleziona una posizione in cui salvare il progetto e premi il pulsante di creazione per creare l'applicazione vuota.

Dopo aver creato l'applicazione di base, dovrai installare i pacchetti richiesti per l'app di esempio.

### Installazione delle dipendenze richieste

1. In Xcode, fai clic con il pulsante destro del mouse sul progetto e scegli **Aggiungi pacchetti...**. Si aprirà la finestra **Pacchetti**, dove potrai aggiungere pacchetti al tuo progetto.
2. Nella finestra **Pacchetti**, aggiungi i seguenti pacchetti:
  - **Per il pacchetto nativo Maplibre, usa questo URL:** <https://github.com/maplibre/maplibre-gl-native-distribution> Dall'URL, aggiungi questi pacchetti: `maplibre-gl-native-distribution`, e `Mapbox`
  - **Per l'SDK iOS per l'autenticazione Amazon Location, usa questo URL:** <https://github.com/aws-geospatial/amazon-location-mobile-auth-sdk-ios>. Dall'URL, aggiungi questi pacchetti:, e `amazon-location-mobile-auth-sdk-ios` `AmazonLocationiOSAuthSDK`
  - **Per l'SDK iOS di Amazon Location Tracking, usa questo URL:** <https://github.com/aws-geospatial/amazon-location-mobile-tracking-sdk-ios>. Dall'URL, aggiungi questi pacchetti:, e `amazon-location-mobile-tracking-sdk-ios` `AmazonLocationIOSTrackingSDK`

## Impostazione del codice iniziale

Abilita le autorizzazioni di localizzazione nella tua app

1. Apri il tuo progetto Xcode.
2. Individua il `Info.plist` file del progetto.
3. Aggiungi le chiavi necessarie per le autorizzazioni di localizzazione in base ai requisiti dell'app. Ecco le chiavi:
  - `NSLocationWhenInUseUsageDescription`: Descrizione del motivo per cui l'app necessita dell'accesso alla posizione quando è in uso.
  - `NSLocationAlwaysAndWhenInUseUsageDescription`: Descrizione del motivo per cui l'app necessita di un accesso continuo alla posizione.

Ora dovrai configurare i valori delle risorse nella tua app. Aggiungi un nuovo file denominato `Config.xcconfig` e compila i valori che avevi creato in precedenza nella console Amazon.

```
REGION =
INDEX_NAME =
MAP_NAME =
IDENTITY_POOL_ID =
TRACKER_NAME =
```

1. Dalla sezione del navigatore sul lato sinistro, seleziona il progetto.
2. Nella sezione obiettivi, seleziona la tua app e fai clic sulla scheda delle informazioni.
3. Aggiungi proprietà informative con valori come i seguenti:
4. Aggiungi il `Config.swift` file con i contenuti seguenti, che leggerà i valori di configurazione dal file di informazioni del pacchetto.

```
import Foundation

enum Config {
    static let region = Bundle.main.object(forKey: "Region") as!
    String
    static let mapName = Bundle.main.object(forKey: "MapName") as!
    String
    static let indexName = Bundle.main.object(forKey: "IndexName")
    as! String
    static let identityPoolId = Bundle.main.object(forKey:
    "IdentityPoolId") as! String
    static let trackerName = Bundle.main.object(forKey:
    "TrackerName") as! String
}
```

5. Crea una nuova cartella con il nome `ViewModel` e aggiungi un `TrackingViewModel.swift` file al suo interno.

```
import SwiftUI
import AmazonLocationiOSAuthSDK
import MapLibre

final class TrackingViewModel : ObservableObject {
    @Published var trackingButtonText = NSLocalizedString("StartTrackingLabel",
    comment: "")
    @Published var trackingButtonColor = Color.blue
}
```

```
@Published var trackingButtonIcon = "play.circle"
@Published var region : String
@Published var mapName : String
@Published var indexName : String
@Published var identityPoolId : String
@Published var trackerName : String
@Published var showAlert = false
@Published var alertTitle = ""
@Published var alertMessage = ""
@Published var centerLabel = ""

var clientIntialised: Bool
var client: LocationTracker!
var authHelper: AuthHelper
var credentialsProvider: LocationCredentialsProvider?
var mlnMapView: MLNMapView?
var mapViewDelegate: MapViewDelegate?
var lastGetTrackingTime: Date?
var trackingActive: Bool

init(region: String, mapName: String, indexName: String, identityPoolId:
String, trackerName: String) {
    self.region = region
    self.mapName = mapName
    self.indexName = indexName
    self.identityPoolId = identityPoolId
    self.trackerName = trackerName
    self.authHelper = AuthHelper()
    self.trackingActive = false
    self.clientIntialised = false
}

func authWithCognito(identityPoolId: String?) {
    guard let identityPoolId =
identityPoolId?.trimmingCharacters(in: .whitespacesAndNewlines)
    else {
        alertTitle = NSLocalizedString("Error", comment: "")
        alertMessage = NSLocalizedString("NotAllFieldsAreConfigured", comment:
"")
        showAlert = true
        return
    }
    credentialsProvider =
authHelper.authenticateWithCognitoUserPool(identityPoolId: identityPoolId)
```

```
        initializeClient()
    }

    func initializeClient() {
        client = LocationTracker(provider: credentialsProvider!, trackerName:
trackerName)
        clientIntialised = true
    }
}
```

## Aggiungere una mappa interattiva all'applicazione

Ora aggiungerai il controllo della mappa alla tua applicazione. Questo tutorial utilizza MapLibre l' AWS API per gestire la visualizzazione della mappa nell'applicazione. Il controllo della mappa stesso fa parte della libreria [MapLibre GL Native iOS](#).

1. Aggiungi `MapView.swift` il file nella cartella Views con il seguente codice:

```
import SwiftUI
import MapLibre

struct MapView: UIViewRepresentable {
    var onMapViewAvailable: ((MLNMapView) -> Void)?
    var mlnMapView: MLNMapView?
    var trackingViewModel: TrackingViewModel

    func makeCoordinator() -> MapView.Coordinator {
        return Coordinator(self, trackingViewModel: trackingViewModel)
    }

    func makeUIView(context: Context) -> MLNMapView {
        let styleURL = URL(string: "https://maps.geo.
\\(trackingViewModel.region).amazonaws.com/maps/v0/maps/
\\(trackingViewModel.mapName)/style-descriptor")
        let mapView = MLNMapView(frame: .zero, styleURL: styleURL)
        mapView.autoresizingMask = [.flexibleWidth, .flexibleHeight]
        mapView.setZoomLevel(15, animated: true)
        mapView.showsUserLocation = true
        mapView.userTrackingMode = .follow
        context.coordinator.mlnMapView = mapView
        mapView.delegate = context.coordinator
    }
}
```

```
        mapView.logoView.isHidden = true
        context.coordinator.addCenterMarker()

        onMapViewAvailable?(mapView)
        trackingViewModel.mlnMapView = mapView
        return mapView
    }

    func updateUIView(_ uiView: MLNMapView, context: Context) {
    }

    class Coordinator: NSObject, MLNMapViewDelegate, MapViewDelegate {
        var control: MapView
        var mlnMapView: MLNMapView?
        var trackingViewModel: TrackingViewModel
        var centerMarker: MLNPointAnnotation?

        public init(_ control: MapView, trackingViewModel: TrackingViewModel) {
            self.control = control
            self.trackingViewModel = trackingViewModel
            super.init()
            self.trackingViewModel.mapViewDelegate = self
        }

        func mapViewDidFinishRenderingMap(_ mapView: MLNMapView, fullyRendered:
Bool) {
            if(fullyRendered) {
                mapView.accessibilityIdentifier = "MapView"
                mapView.isAccessibilityElement = false
            }
        }

        func addCenterMarker() {
            guard let mlnMapView = mlnMapView else {
                return
            }

            let centerCoordinate = mlnMapView.centerCoordinate
            let marker = MLNPointAnnotation()
            marker.coordinate = centerCoordinate
            marker.accessibilityLabel = "CenterMarker"
            mlnMapView.addAnnotation(marker)
            centerMarker = marker
        }
    }
}
```



```

        trackingViewModel.reverseGeocodeCenter(centerCoordinate:
centerCoordinate, marker: marker)
    }

    func mapView(_ mapView: MLNMapView, regionDidChangeAnimated animated: Bool)
    {
        if let marker = centerMarker {
            DispatchQueue.main.asyncAfter(deadline: .now() + 1.0)
            {
                mapView.deselectAnnotation(marker, animated: false)
                marker.coordinate = mapView.centerCoordinate
                let centerCoordinate = mapView.centerCoordinate
                self.trackingViewModel.reverseGeocodeCenter(centerCoordinate:
centerCoordinate, marker: marker)
            }
        }
    }
}

```

2. Aggiungi `AWSSignatureV4Delegate` il file nella `ViewModel` cartella. Questo file viene utilizzato per firmare con tutte le richieste `MapView` http per il rendering della mappa:

```

import MapLibre
import AmazonLocationiOSAuthSDK

class AWSSignatureV4Delegate : NSObject, MLNOfflineStorageDelegate {
    private let awsSigner: AWSSigner

    init(credentialsProvider: LocationCredentialsProvider) {
        self.awsSigner = DENY LIST ERROR , serviceName: "geo")
        super.init()
    }

    func offlineStorage(_ storage: MLNOfflineStorage, urlForResourceOf kind:
MLNResourceKind, with url: URL) -> URL {
        if url.host?.contains("amazonaws.com") != true {
            return url
        }
        let signedURL = awsSigner.signURL(url: url, expires: .hours(1))

        return signedURL
    }
}

```

```
}
```

3. Aggiungi `UserLocationView.swift` il file nella cartella `Views`. Questo aggiunge un pulsante che centra la mappa sulla posizione dell'utente

```
import SwiftUI

struct UserLocationView: View {
    @ObservedObject var trackingViewModel: TrackingViewModel
    var body: some View {
        Button(action: {
            trackingViewModel.locateMe()
        }) {
            Image(systemName: "scope")
                .resizable()
                .frame(width: 24, height: 24)
                .padding(5)
                .background(Color.white)
                .foregroundColor(.blue)
                .clipShape(RoundedRectangle(cornerRadius: 8))
                .shadow(color: Color.black.opacity(0.3), radius: 3, x: 0, y: 2)
        }
        .accessibility(identifier: "LocateMeButton")
        .padding(.trailing, 10)
        .padding(.bottom, 10)
        .frame(maxWidth: .infinity, alignment: .trailing)
    }
}
```

4. Aggiungi il `TrackingView.swift` file con il seguente codice:

```
import SwiftUI

struct TrackingView: View {
    @ObservedObject var trackingViewModel: TrackingViewModel
    var body: some View {
        ZStack(alignment: .bottom) {
            MapView(trackingViewModel: trackingViewModel)
            VStack {
                UserLocationView(trackingViewModel: trackingViewModel)
            }
        }
        .onAppear() {
```

```
        if !trackingViewModel.identityPoolId.isEmpty {
            trackingViewModel.authWithCognito(identityPoolId:
trackingViewModel.identityPoolId)
        }
    }
}
```

Ora puoi creare l'applicazione. Per eseguirla, potresti dover configurare un dispositivo per emularla in Xcode o utilizzare l'app sul tuo dispositivo. Usa questa app per vedere come si comporta il controllo della mappa. Puoi effettuare una panoramica trascinandola sulla mappa e pizzicandola per ingrandirla. Da soli, puoi modificare il funzionamento del controllo della mappa per personalizzarlo in base alle esigenze della tua applicazione.

## Aggiungere la ricerca all'applicazione

Ora aggiungerai la ricerca con geocodifica inversa all'applicazione, in cui troverai gli elementi in una posizione. Per semplificare l'uso di un'app iOS, cercheremo al centro dello schermo. Per trovare una nuova posizione, sposta la mappa nel punto in cui desideri effettuare la ricerca. Posizioneremo un indicatore al centro della mappa per mostrare dove stiamo cercando.

1. Aggiungi il seguente codice nel file `TrackingViewModel.swift` che è correlato alla ricerca con geocodifica inversa

```
func reverseGeocodeCenter(centerCoordinate: CLLocationCoordinate2D, marker:
MLNPointAnnotation) {
    let position = [NSNumber(value: centerCoordinate.longitude), NSNumber(value:
centerCoordinate.latitude)]
    searchPositionAPI(position: position, marker: marker)
}

func searchPositionAPI(position: [Double], marker: MLNPointAnnotation) {
    if let amazonClient = authHelper.getLocationClient() {
        Task {
            let searchRequest = SearchPlaceIndexForPositionInput(indexName:
indexName, language: "en" , maxResults: 10, position: position)
            let searchResponse = try? await amazonClient.searchPosition(indexName:
indexName, input: searchRequest)
            DispatchQueue.main.async {
                self.centerLabel = searchResponse?.results?.first?.place?.label ??
""
            }
        }
    }
}
```

```

        self.mlnMapView?.selectAnnotation(marker, animated: true,
completionHandler: {})
    }
}
}
}

```

2. Aggiorna `TrackingView.swift` il file con il seguente codice che mostrerà l'indirizzo della posizione centrata di mapview

```

import SwiftUI

struct TrackingView: View {
    @ObservedObject var trackingViewModel: TrackingViewModel
    var body: some View {
        ZStack(alignment: .bottom) {
            if trackingViewModel.mapSigningIntialised {
                MapView(trackingViewModel: trackingViewModel)
                VStack {
                    UserLocationView(trackingViewModel: trackingViewModel)
                    CenterAddressView(trackingViewModel: trackingViewModel)
                }
            }
            else {
                Text("Loading...")
            }
        }
        .onAppear() {
            if !trackingViewModel.identityPoolId.isEmpty {
                Task {
                    do {
                        try await trackingViewModel.authWithCognito(identityPoolId:
trackingViewModel.identityPoolId)
                    }
                    catch {
                        print(error)
                    }
                }
            }
        }
    }
}

```

## Aggiungere il tracciamento alla tua applicazione

L'ultimo passaggio per l'applicazione consiste nell'aggiungere funzionalità di tracciamento all'app. In questo caso, aggiungerai start tracking, stop tracking, recupererà e mostrerà i tracker point sulla tua app.

1. Aggiungi il `TrackingBottomView.swift` file nel tuo progetto. Che ha un pulsante che avvia e interrompe il tracciamento delle posizioni degli utenti e mostra i punti di tracciamento sulla mappa.

```
import SwiftUI

struct TrackingBottomView: View {
    @ObservedObject var trackingViewModel: TrackingViewModel
    var body: some View {
        Button(action: {
            Task {
                if(trackingViewModel.trackingButtonText ==
NSLocalizedString("StartTrackingLabel", comment: "")) {
                    trackingViewModel.startTracking()
                } else {
                    trackingViewModel.stopTracking()
                }
            }
        }) {
            HStack {
                Spacer()
                Text("Tracking")
                    .foregroundColor(trackingViewModel.trackingButtonColor)
                    .background(.white)
                    .cornerRadius(15.0)

                Image(systemName: trackingViewModel.trackingButtonIcon)
                    .resizable()
                    .frame(width: 24, height: 24)
                    .padding(5)
                    .background(.white)
                    .foregroundColor(trackingViewModel.trackingButtonColor)
            }
        }
        .accessibility(identifier: "TrackingButton")
    }
}
```

```
        .background(.white)
        .clipShape(RoundedRectangle(cornerRadius: 8))
        .padding(.trailing, 10)
        .padding(.bottom, 40)
        .frame(width: 130, alignment: .trailing)
        .shadow(color: Color.black.opacity(0.3), radius: 3, x: 0, y: 2)
    }
}
```

## 2. Aggiorna TrackingView.swift il file con il seguente codice

```
import SwiftUI

struct TrackingView: View {
    @ObservedObject var trackingViewModel: TrackingViewModel
    var body: some View {
        ZStack(alignment: .bottom) {
            if trackingViewModel.mapSigningInitialised {
                MapView(trackingViewModel: trackingViewModel)
                VStack {
                    UserLocationView(trackingViewModel: trackingViewModel)
                    CenterAddressView(trackingViewModel: trackingViewModel)
                    TrackingBottomView(trackingViewModel: trackingViewModel)
                }
            }
            else {
                Text("Loading...")
            }
        }
        .onAppear() {
            if !trackingViewModel.identityPoolId.isEmpty {
                Task {
                    do {
                        try await trackingViewModel.authWithCognito(identityPoolId:
trackingViewModel.identityPoolId)
                    }
                    catch {
                        print(error)
                    }
                }
            }
        }
    }
}
```

```
}
```

3. Aggiungi il seguente codice nel `TrackingViewModel.swift` file. Queste funzioni sono responsabili dell'avvio e dell'arresto del tracciamento. Mostrerà anche un avviso di errore se l'autorizzazione alla localizzazione dell'utente viene negata.
4. Per implementare il tracciamento in primo piano, copia e incolla il seguente esempio di codice:

```
func showLocationDeniedRazionale() {
    alertTitle = NSLocalizedString("locationManagerAlertTitle", comment: "")
    alertMessage = NSLocalizedString("locationManagerAlertText", comment: "")
    showAlert = true
}

// Required in info.plist: Privacy - Location When In Use Usage Description
func startTracking() {
    do {
        print("Tracking Started...")
        if(client == nil) {
            initializeClient()
        }
        try client.startTracking()
        DispatchQueue.main.async { [self] in
            self.trackingButtonText = NSLocalizedString("StopTrackingLabel",
comment: "")
            self.trackingButtonColor = .red
            self.trackingButtonIcon = "pause.circle"
            trackingActive = true
        }
    } catch TrackingLocationError.permissionDenied {
        showLocationDeniedRazionale()
    } catch {
        print("error in tracking")
    }
}

func stopTracking() {
    print("Tracking Stopped...")
    client.stopTracking()
    trackingButtonText = NSLocalizedString("StartTrackingLabel", comment: "")
    trackingButtonColor = .blue
    trackingButtonIcon = "play.circle"
    trackingActive = false
}
```

}

**Note**

`startTracking` Chiederà l'autorizzazione alla localizzazione dell'utente. L'applicazione deve utilizzare le autorizzazioni `When In Use` o `Only Once`. In caso contrario, l'applicazione genererà un errore di autorizzazione negata.

Per ottenere e visualizzare le posizioni di tracciamento, segui questa procedura:

1. Per ottenere le posizioni dal dispositivo dell'utente, devi fornire la data e l'ora di inizio e fine. Una singola chiamata restituisce un massimo di 100 posizioni di tracciamento, ma se ci sono più di 100 posizioni di tracciamento, restituirà un valore `nextToken`. Dovrai chiamare le successive chiamate `getTrackerDeviceLocation` con `NextToken` per caricare più punti di tracciamento per l'ora di inizio e di fine specificata.

```
func getTrackingPoints(nextToken: String? = nil) async throws {
    guard trackingActive else {
        return
    }
    // Initialize startTime to 24 hours ago from the current date and time.
    let startTime: Date = Date().addingTimeInterval(-86400)
    var endTime: Date = Date()
    if lastGetTrackingTime != nil {
        endTime = lastGetTrackingTime!
    }
    let result = try await client?.getTrackerDeviceLocation(nextToken:
nextToken, startTime: startTime, endTime: endTime)
    if let trackingData = result {

        lastGetTrackingTime = Date()
        let devicePositions = trackingData.devicePositions

        let positions = devicePositions!.sorted { (pos1:
LocationClientTypes.DevicePosition, pos2: LocationClientTypes.DevicePosition) ->
Bool in
            guard let date1 = pos1.sampleTime,
                  let date2 = pos2.sampleTime else {
                return false
            }
    }
}
```



```

        return date1 < date2
    }

    let trackingPoints = positions.compactMap { position ->
CLLocationCoordinate2D? in
        guard let latitude = position.position!.last, let longitude =
position.position!.first else {
            return nil
        }
        return CLLocationCoordinate2D(latitude: latitude, longitude:
longitude)
    }
    DispatchQueue.main.async {
        self.mapViewDelegate!.drawTrackingPoints( trackingPoints:
trackingPoints)
    }
    if let nextToken = trackingData.nextToken {
        try await getTrackingPoints(nextToken: nextToken)
    }
}
}

```

## 2. Ora sostituisci il codice nel file con il seguente codice: MapView.swift

```

import SwiftUI
import MapLibre

struct MapView: UIViewRepresentable {
    var onMapViewAvailable: ((MLNMapView) -> Void)?
    var mlnMapView: MLNMapView?
    var trackingViewModel: TrackingViewModel

    func makeCoordinator() -> MapView.Coordinator {
        return Coordinator(self, trackingViewModel: trackingViewModel)
    }

    func makeUIView(context: Context) -> MLNMapView {
        let styleURL = URL(string: "https://maps.geo.
\\(trackingViewModel.region).amazonaws.com/maps/v0/maps/
\\(trackingViewModel.mapName)/style-descriptor")
        let mapView = MLNMapView(frame: .zero, styleURL: styleURL)
        mapView.autoresizingMask = [.flexibleWidth, .flexibleHeight]
        mapView.setZoomLevel(15, animated: true)
        mapView.showsUserLocation = true
    }
}

```

```
mapView.userTrackingMode = .follow
context.coordinator.mlnMapView = mapView
mapView.delegate = context.coordinator

mapView.logoView.isHidden = true
context.coordinator.addCenterMarker()

onMapViewAvailable?(mapView)
trackingViewModel.mlnMapView = mapView
return mapView
}

func updateUIView(_ uiView: MLNMapView, context: Context) {
}

class Coordinator: NSObject, MLNMapViewDelegate, MapViewDelegate {
    var control: MapView
    var mlnMapView: MLNMapView?
    var trackingViewModel: TrackingViewModel
    var centerMarker: MLNPointAnnotation?

    public init(_ control: MapView, trackingViewModel: TrackingViewModel) {
        self.control = control
        self.trackingViewModel = trackingViewModel
        super.init()
        self.trackingViewModel.mapViewDelegate = self
    }

    func mapViewDidFinishRenderingMap(_ mapView: MLNMapView, fullyRendered:
Bool) {
        if(fullyRendered) {
            mapView.accessibilityIdentifier = "MapView"
            mapView.isAccessibilityElement = false
        }
    }

    func addCenterMarker() {
        guard let mlnMapView = mlnMapView else {
            return
        }

        let centerCoordinate = mlnMapView.centerCoordinate
        let marker = MLNPointAnnotation()
        marker.coordinate = centerCoordinate
    }
}
```

```
marker.accessibilityLabel = "CenterMarker"
mLnMapView.addAnnotation(marker)
centerMarker = marker

trackingViewModel.reverseGeocodeCenter(centerCoordinate:
centerCoordinate, marker: marker)
}

func mapView(_ mapView: MLNMapView, regionDidChangeAnimated animated: Bool)
{
    if let marker = centerMarker {
        DispatchQueue.main.asyncAfter(deadline: .now() + 1.0) {
            mapView.deselectAnnotation(marker, animated: false)
            marker.coordinate = mapView.centerCoordinate
            let centerCoordinate = mapView.centerCoordinate
            self.trackingViewModel.reverseGeocodeCenter(centerCoordinate:
centerCoordinate, marker: marker)
        }
    }
}

func mapView(_ mapView: MLNMapView, viewFor annotation: MLNAnnotation) ->
MLNAnnotationView? {
    guard let pointAnnotation = annotation as? MLNPointAnnotation else {
        return nil
    }

    let reuseIdentifier: String
    var color: UIColor = .black
    if pointAnnotation.accessibilityLabel == "Tracking" {
        reuseIdentifier = "TrackingAnnotation"
        color = UIColor(red: 0.00784313725, green: 0.50588235294, blue:
0.58039215686, alpha: 1)
    } else if pointAnnotation.accessibilityLabel == "LocationChange" {
        reuseIdentifier = "LocationChange"
        color = .gray
    } else {
        reuseIdentifier = "DefaultAnnotationView"
    }

    var annotationView =
mapView.dequeueReusableAnnotationView(withIdentifier: reuseIdentifier)

    if annotationView == nil {
```

```

        if reuseIdentifier != "DefaultAnnotationView" {
            annotationView = MLNAnnotationView(annotation: annotation,
reuseIdentifier: reuseIdentifier)
            //If point annotation is an uploaded Tracking point the radius
is 20 and color is blue, otherwise radius is 10 and color is gray
            let radius = pointAnnotation.accessibilityLabel == "Tracking" ?
20:10

            annotationView?.frame = CGRect(x: 0, y: 0, width: radius,
height: radius)

            annotationView?.backgroundColor = color
            annotationView?.layer.cornerRadius = 10

            if pointAnnotation.accessibilityLabel == "Tracking" {
                annotationView?.layer.borderColor = UIColor.white.cgColor
                annotationView?.layer.borderWidth = 2.0
                annotationView?.layer.shadowColor = UIColor.black.cgColor
                annotationView?.layer.shadowOffset = CGSize(width: 0,
height: 2)

                annotationView?.layer.shadowRadius = 3
                annotationView?.layer.shadowOpacity = 0.2
                annotationView?.clipsToBounds = false
            }
        }
        else {
            return nil
        }
    }

    return annotationView
}

func mapView(_ mapView: MLNMapView, didUpdate userLocation:
MLNUserLocation?) {
    if (userLocation?.location) != nil {
        if trackingViewModel.trackingActive {
            let point = MLNPointAnnotation()
            point.coordinate = (userLocation?.location!.coordinate)!
            point.accessibilityLabel = "LocationChange"
            mapView.addAnnotation(point)
            Task {
                do {
                    try await trackingViewModel.getTrackingPoints()
                }
                catch {

```

```

        print(error)
    }
}
}

func checkIfTrackingAnnotationExists(on mapView: MLNMapView, at
coordinates: CLLocationCoordinate2D) -> Bool {
    let existingAnnotation = mapView.annotations?.first(where: { annotation
in
        guard let annotation = annotation as? MLNPointAnnotation else
{ return false }
        return annotation.coordinate.latitude == coordinates.latitude &&
        annotation.coordinate.longitude == coordinates.longitude &&
        annotation.accessibilityLabel == "Tracking" })
    return existingAnnotation != nil
}

public func drawTrackingPoints(trackingPoints: [CLLocationCoordinate2D]?) {
    guard let mapView = mlnMapView, let newTrackingPoints =
trackingPoints, !newTrackingPoints.isEmpty else {
        return
    }

    let uniqueCoordinates = newTrackingPoints.filter { coordinate in
        !checkIfTrackingAnnotationExists(on: mapView, at: coordinate)
    }

    let points = uniqueCoordinates.map { coordinate -> MLNPointAnnotation
in
        let point = MLNPointAnnotation()
        point.coordinate = coordinate
        point.accessibilityLabel = "Tracking"
        return point
    }
    mapView.addAnnotations(points)
}
}

protocol MapViewDelegate: AnyObject {
    func drawTrackingPoints(trackingPoints: [CLLocationCoordinate2D]?)

```

```
}
```

Per localizzare i valori delle stringhe, utilizzate la procedura seguente.

1. Crea e aggiungi un nuovo file chiamato `Localizable.xcstrings`.
2. Fai clic con il pulsante destro del mouse sul `Localizable.xcstrings` file e aprilo come codice sorgente.
3. Sostituisci il suo contenuto con quanto segue:

```
{
  "sourceLanguage" : "en",
  "strings" : {
    "Cancel" : {
      "extractionState" : "manual",
      "localizations" : {
        "en" : {
          "stringUnit" : {
            "state" : "translated",
            "value" : "Cancel"
          }
        }
      }
    },
    "Error" : {
      "extractionState" : "manual",
      "localizations" : {
        "en" : {
          "stringUnit" : {
            "state" : "translated",
            "value" : "Error"
          }
        }
      }
    },
    "Loading..." : {

  },
  "locationManagerAlertText" : {
    "extractionState" : "manual",
    "localizations" : {
      "en" : {
```

```
        "stringUnit" : {
            "state" : "translated",
            "value" : "Allow \\\\"Quick Start App\\" to use your location"
        }
    }
},
"locationManagerAlertTitle" : {
    "extractionState" : "manual",
    "localizations" : {
        "en" : {
            "stringUnit" : {
                "state" : "translated",
                "value" : "We need your location to detect your location in map"
            }
        }
    }
},
"NotAllFieldsAreConfigured" : {
    "extractionState" : "manual",
    "localizations" : {
        "en" : {
            "stringUnit" : {
                "state" : "translated",
                "value" : "Not all the fields are configured"
            }
        }
    }
},
"OK" : {
    "extractionState" : "manual",
    "localizations" : {
        "en" : {
            "stringUnit" : {
                "state" : "translated",
                "value" : "OK"
            }
        }
    }
},
"StartTrackingLabel" : {
    "localizations" : {
        "en" : {
            "stringUnit" : {
```

```
        "state" : "translated",
        "value" : "Start Tracking"
    }
}
},
"StopTrackingLabel" : {
    "localizations" : {
        "en" : {
            "stringUnit" : {
                "state" : "translated",
                "value" : "Stop Tracking"
            }
        }
    }
},
"Tracking" : {
}
},
"version" : "1.0"
}
```

4. Salva i tuoi file e crea ed esegui l'app per visualizzare in anteprima la funzionalità.
5. Consenti l'autorizzazione alla posizione e tocca il pulsante di tracciamento. L'app inizierà a caricare le posizioni degli utenti e le caricherà sul localizzatore di localizzazione Amazon. Mostrerà anche le modifiche alla posizione dell'utente, i punti di tracciamento e l'indirizzo corrente sulla mappa.

La tua richiesta di avvio rapido è completa. Questo tutorial ti ha mostrato come creare un'applicazione iOS che:

- Crea una mappa con cui gli utenti possono interagire.
- Gestisce diversi eventi della mappa associati alla modifica della visualizzazione della mappa da parte dell'utente.
- Richiama un'API Amazon Location Service, in particolare per cercare sulla mappa di una posizione, utilizzando l' `searchByPosition` API di Amazon Location.



## Cosa c'è dopo

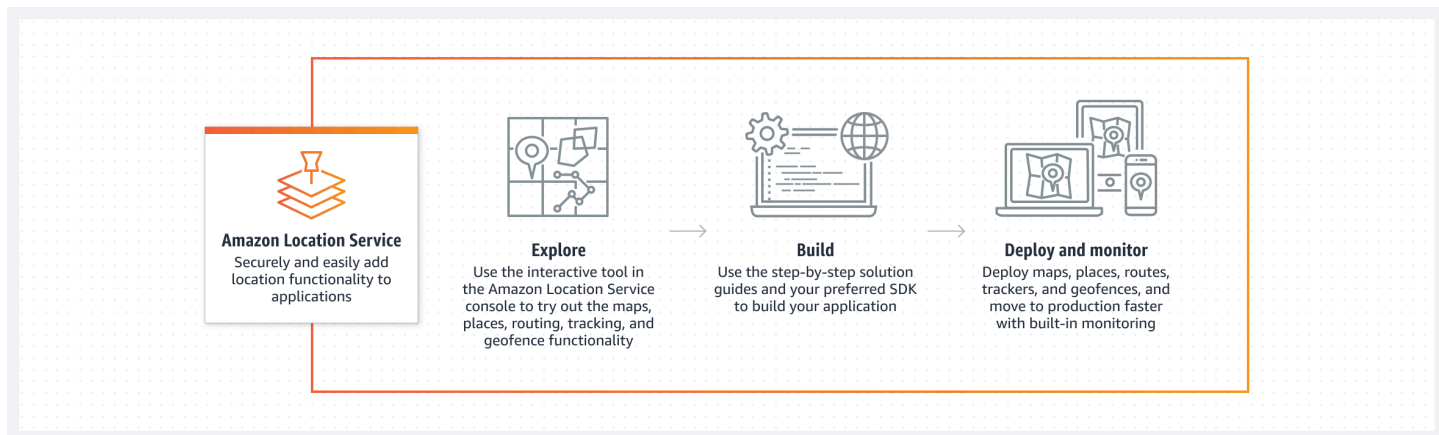
Il codice sorgente di questa applicazione è disponibile su [GitHub](#).

Per ottenere di più da Amazon Location, puoi consultare le seguenti risorse:

- Approfondisci i [concetti di Amazon Location Service](#)
- Ottieni ulteriori informazioni su [come utilizzare le caratteristiche e le funzionalità di Amazon Location](#)
- Scopri come ampliare questo esempio e creare applicazioni più complesse esaminando [esempi di codice con Amazon Location](#)

# Concetti di Amazon Location Service

Con Amazon Location Service, puoi aggiungere dati sulla posizione alla tua applicazione in modo sicuro. Esplora alcune funzionalità utilizzando lo [strumento visivo e interattivo](#), disponibile sulla console Amazon Location. Utilizzando lo strumento di esplorazione, puoi manipolare una mappa predefinita, cercare punti di interesse, disegnare geofence attorno alle aree di interesse e simulare l'invio delle posizioni dei dispositivi a un tracker.



Quando sei pronto per creare, crea le tue risorse e scegli tra una varietà di stili di mappe e fornitori di dati. Quindi puoi installare l'SDK adatto al tuo ambiente di sviluppo e utilizzare le API di Amazon Location seguendo le istruzioni contenute in questa guida. Inoltre, puoi integrare il monitoraggio utilizzando Amazon CloudWatch e AWS CloudTrail.

Gli argomenti di questa sezione forniscono una panoramica dei concetti fondamentali di Amazon Location e ti preparano a iniziare a utilizzare la posizione nelle tue applicazioni.

## Argomenti

- [Panoramica delle sedi Amazon](#)
- [Mappe](#)
- [Ricerca di luoghi](#)
- [Route](#)
- [Geofence e tracker](#)
- [Casi d'uso comuni per l'utilizzo di Amazon Location Service](#)
- [Cos'è un fornitore di dati?](#)
- [Regioni di localizzazione ed endpoint di Amazon](#)
- [Quote di Amazon Location Service](#)

## Panoramica delle sedi Amazon

Amazon Location Service fornisce l'accesso a funzionalità e fornitori di dati basati sulla posizione tramite AWS risorse. Amazon Location offre cinque tipi di AWS risorse, a seconda del tipo di funzionalità di cui hai bisogno. Usa le diverse risorse insieme per creare un'applicazione completa basata sulla posizione. Puoi creare una o più di queste risorse utilizzando la console Amazon Location, le API di Amazon Location o gli SDK.

Ogni risorsa definisce il [fornitore di dati](#) sottostante da utilizzare (ove applicabile) e dà accesso a funzionalità correlate al tipo di risorsa.

Per esempio:

- [Amazon Location Service Maps](#) ti consente di scegliere una mappa da un fornitore di mappe da utilizzare sulla tua applicazione mobile o web.
- [Amazon Location Service Places](#) ti consente di scegliere un fornitore di dati per la ricerca di punti di interesse, il completamento di testo parziale, la geocodifica e la geocodifica inversa.
- [Amazon Location Service Routes](#) ti consente di scegliere un fornitore di dati, trovare percorsi e stimare il tempo di viaggio in base alle informazioni up-to-date stradali e sul traffico in tempo reale.
- [Amazon Location Service Geofences](#) ti consente di definire le aree di interesse come confini virtuali. Potrai quindi valutare le posizioni rispetto ad esse e ricevere notifiche sugli eventi di entrata e uscita.
- I [localizzatori di Amazon Location Service](#) ricevono aggiornamenti sulla posizione dai tuoi dispositivi. Puoi collegare i tracker alle raccolte di geofence in modo che tutti gli aggiornamenti di posizione vengano valutati automaticamente rispetto ai geofence.

Puoi utilizzare le policy IAM per gestire e autorizzare l'accesso alle tue risorse Amazon Location. Puoi anche organizzare le tue risorse in gruppi di risorse per gestire e automatizzare le attività man mano che il numero delle risorse aumenta. Per ulteriori informazioni sulla gestione AWS delle risorse, consulta [What are AWS Resource Groups?](#) Nella AWS Resource Groups User Guide.

La posizione viene definita utilizzando coordinate di latitudine e longitudine che seguono il [World Geodetic System \(WGS 84\)](#), comunemente usato come sistema di riferimento di coordinate standard per i servizi GPS (Global Positioning System).

Le seguenti sezioni descrivono come funzionano i componenti di Amazon Location.

# Mappe

La risorsa Amazon Location Service Map ti consente di accedere ai dati della mappa di base sottostanti per una mappa. Utilizza la risorsa Map con una libreria di rendering delle mappe per aggiungere una mappa interattiva alla tua applicazione. È possibile aggiungere altre funzionalità alla mappa, come marker (o pin), percorsi e aree poligonali, in base alle esigenze dell'applicazione.

## Note

Per informazioni su come utilizzare in pratica le risorse cartografiche, consulta [Utilizzo di Amazon Location Maps nella tua applicazione](#)

Di seguito è riportata una panoramica su come creare e utilizzare le risorse cartografiche:

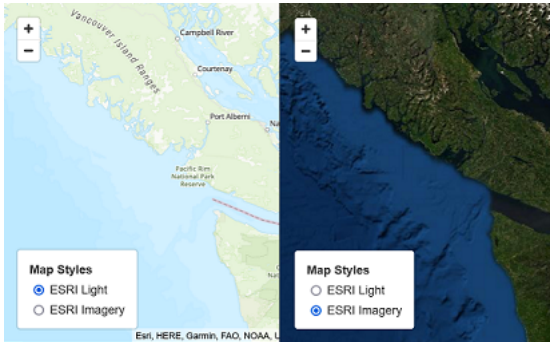


1. Puoi creare una risorsa cartografica nel tuo AWS account selezionando uno stile di mappa da un fornitore di dati.
2. Puoi quindi selezionare e installare l'SDK più adatto al tuo ambiente di sviluppo e alle tue applicazioni. Per ulteriori informazioni sulle opzioni disponibili, consulta l'argomento sull'[accesso ad Amazon Location](#).
3. Per visualizzare una mappa nell'applicazione, combinate una risorsa cartografica con una libreria di rendering, come Amplify MapLibre o Tangram. Per ulteriori informazioni, consulta [Uso delle mappe](#) in questa guida.
4. Puoi quindi integrare il monitoraggio utilizzando servizi come Amazon CloudWatch e AWS CloudTrail Amazon Location. Per ulteriori informazioni, consulta [Monitoraggio di Amazon Location Service con Amazon CloudWatch](#) e [Registrazione e monitoraggio con AWS CloudTrail](#).

## Stili di mappa

Quando si crea una risorsa cartografica, è necessario scegliere uno stile di mappa per quella risorsa. Gli stili di mappa definiscono l'aspetto della mappa renderizzata. Ad esempio, l'immagine seguente

mostra lo stesso fornitore di dati con due stili diversi provenienti da diverse risorse cartografiche in Amazon Location. Uno stile è uno stile stradale tipico, basato sui dati vettoriali nella mappa. L'altro include dati raster che mostrano immagini satellitari. Lo stile può cambiare man mano che si ingrandisce o rimpicciolisce la mappa, ma in genere gli stili hanno un tema coerente. È possibile sovrascrivere parti o tutte le informazioni sullo stile prima di passarle alla libreria di rendering delle mappe.



## Opinioni politiche

Alcuni stili di mappe in Amazon Location Service supportano opinioni politiche aggiuntive.

### Note

L'opinione politica deve essere utilizzata in conformità con le leggi applicabili, comprese quelle sulla mappatura del paese o della regione in cui sono disponibili mappe, immagini e altri dati e contenuti di terze parti a cui accedi tramite Amazon Location Service.

I seguenti stili di mappa supportano una visione politica dell'India (IND).

- [Stili di mappa Esri:](#)
  - Navigazione Esri
  - Luce Esri
  - Mappa stradale di Esri
  - Tela Esri grigio scuro
  - Tela Esri grigio chiaro
- [Stili di mappa Open Data:](#)
  - Open Data Standard Light
  - Open Data Standard Dark

- Open Data Visualization Light
- Open Data Visualization Dark

Nella console Amazon Location Service, puoi filtrare gli stili mostrati per mostrare solo gli stili che supportano la visione politica dell'India.

## Livelli personalizzati

Un livello personalizzato è un livello aggiuntivo che puoi abilitare per uno stile di mappa. Attualmente solo lo stile della VectorEsriNavigation mappa supporta il livello POI personalizzato.

Quando abiliti il layer POI personalizzato, aggiunge alla mappa un set più ricco di luoghi, come negozi, servizi, ristoranti, attrazioni e altri punti di interesse. Per impostazione predefinita, il livello personalizzato è unset. Per ulteriori informazioni, consulta il riferimento [MapConfiguration](#) all'API Location.

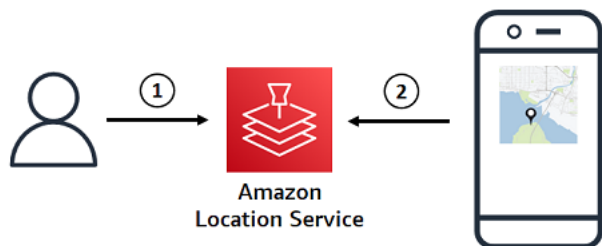
## Rendering della mappa

Per eseguire il rendering di una mappa nell'applicazione, in genere si utilizza una libreria di rendering delle mappe. Esistono diverse opzioni comuni da utilizzare per le librerie:

- MapLibre— MapLibre è una libreria open source specifica per il rendering di mappe interattive ed è il metodo preferito per il rendering di mappe da Amazon Location Service. MapLibre include la possibilità di eseguire il rendering di dati raster e vettoriali da una fonte di dati (come una risorsa della mappa di Amazon Location). Puoi estendere MapLibre per disegnare i tuoi dati sulla mappa.
- Amplify — Amplify è un framework open source per la creazione di applicazioni per il Web, iOS, Android e altro ancora. Se la tua applicazione utilizza Amplify, puoi estenderla per includere la funzionalità Amazon Location. Amplify include librerie specifiche per la creazione di applicazioni basate su Amazon Location, tra cui mappe di rendering. Amplify MapLibre utilizza per renderizzare la mappa, ma fornisce funzionalità aggiuntive specifiche di Amazon Location Service per renderla più efficiente da usare, e aggiunge anche funzionalità di ricerca e altre funzionalità.
- Tangram: Tangram è una libreria open source alternativa che rende mappe interattive, simili a MapLibre

La libreria di rendering delle mappe estrae i dati da Amazon Location Service in fase di esecuzione, visualizzando i dati della mappa in base alla risorsa cartografica selezionata. La risorsa cartografica definisce il fornitore di dati e lo stile della mappa che verranno utilizzati.

L'immagine seguente mostra come la risorsa cartografica viene utilizzata in Amazon Location Service insieme a una libreria di rendering delle mappe per creare la mappa finale.



1. Crei una risorsa cartografica in Amazon Location Service, utilizzando AWS Management Console o AWS CLI. Questo definisce il fornitore di dati e lo stile di mappa che desideri utilizzare.
2. L'applicazione include una libreria per il rendering delle mappe. Si assegna alla libreria di rendering delle mappe il nome della risorsa cartografica da utilizzare. La libreria per il rendering delle mappe estrae dati e informazioni di stile per quella risorsa cartografica da Amazon Location e visualizza la mappa sullo schermo.

## Terminologia delle mappe

### Risorsa cartografica

Consente di accedere ai dati cartografici di un provider selezionato. Utilizza la risorsa della mappa per recuperare i riquadri della mappa che contengono i dati della mappa e un descrittore di stile per specificare il modo in cui le feature vengono visualizzate su una mappa.

### Mappa di base

Fornisce un contesto geografico alla mappa, che viene memorizzata come layer di tile vettoriali. I Tile Layer includono il contesto geografico come i nomi delle strade, gli edifici e l'uso del suolo come riferimento visivo.

### Vettore

I dati vettoriali sono dati di forma composti da punti, linee e poligoni. Vengono spesso utilizzati per memorizzare e visualizzare strade, luoghi e aree su una mappa. Una forma vettoriale può essere utilizzata anche come icone per gli indicatori su una mappa.

## Raster

I dati raster sono dati di immagine, costituiti da una griglia, generalmente di colori. Vengono spesso utilizzati per archiviare e visualizzare una rappresentazione di dati continui su mappe, come il terreno, le immagini satellitari o le mappe termiche. Le immagini raster possono essere utilizzate anche come immagini o icone.

## Stile della mappa

I dati vettoriali non includono intrinsecamente informazioni su come disegnare i livelli di dati per creare la mappa finale. Uno stile di mappa definisce il colore e altre informazioni di stile dei dati per definire come appariranno una volta renderizzati. Le risorse cartografiche includono informazioni sullo stile della mappa.

Amazon Location Service fornisce stili che seguono le specifiche di [stile di Mapbox GL](#).

## Piastrella vettoriale

Un formato a riquadri che memorizza i dati della mappa utilizzando forme vettoriali. Questi dati si traducono in una mappa che può adattarsi alla risoluzione dello schermo e renderizzare selettivamente le feature in diversi modi, pur mantenendo una dimensione di file ridotta per prestazioni ottimali.

Formato di file vettoriale supportato: Mapbox Vector Tiles (MVT).

## File glifo

Un file binario contenente caratteri Unicode codificati. Utilizzato da un renderer di mappe per visualizzare le etichette.

## File Sprite

Un file di immagine PNG (Portable Network Graphic) che contiene piccole immagini raster, con descrizioni delle posizioni in un file JSON. Utilizzato da un renderer di mappe per renderizzare icone o texture su una mappa.



## Ricerca di luoghi

Una funzione chiave di Amazon Location Service è la possibilità di cercare informazioni di geolocalizzazione. Amazon Location fornisce questa funzionalità tramite la risorsa Place index.

### Note

Per informazioni su come utilizzare le risorse dell'indice dei luoghi per effettuare ricerche pratiche, consulta [Ricerca di dati relativi a località e geolocalizzazione utilizzando Amazon Location](#).

Puoi utilizzare le API dell'indice dei luoghi per cercare:

- Punti di interesse, come ristoranti e monumenti. Effettua la ricerca per nome e, se lo desideri, per visualizzare un elenco di opzioni ordinate in base alla pertinenza.
- Un indirizzo, che riceve una latitudine e una longitudine per quell'indirizzo. Questa operazione è nota come geocodifica.
- Una posizione di latitudine e longitudine, che riceve l'indirizzo stradale associato o altre informazioni sulla posizione. Questa operazione è nota come geocodifica inversa.
- Una query di testo in formato libero parziale o con errori di ortografia, in genere eseguita da un utente. Questa operazione è nota come completamento automatico, suggerimento automatico o corrispondenza fuzzy.

L'indice dei luoghi include il fornitore di dati da utilizzare per la ricerca.

### Note

I dati cartografici e altre informazioni di geolocalizzazione, comprese le posizioni esatte, possono variare a seconda dei fornitori di dati. Come best practice, utilizza lo stesso fornitore di dati per l'indice dei luoghi, la mappa e altre risorse di Amazon Location. Ad esempio, se i luoghi restituiti dall'indice dei luoghi non corrispondono alla posizione degli stessi luoghi forniti dalla risorsa cartografica, puoi posizionare un indicatore in quella che sembra essere la posizione sbagliata sulla mappa.

Di seguito viene illustrato come creare e utilizzare le risorse dell'indice dei luoghi:



1. Innanzitutto, crei una risorsa relativa all'indice dei luoghi nel tuo AWS account selezionando un fornitore di dati.
2. Puoi quindi selezionare e installare l'SDK più adatto al tuo ambiente di sviluppo e alle tue applicazioni. Per ulteriori informazioni sulle opzioni disponibili, consulta l'argomento sull'[accesso ad Amazon Location](#).
3. Inizia a utilizzare le API Amazon Location Places. Per ulteriori informazioni, consulta l'argomento sull'utilizzo della [ricerca in Places](#).
4. È quindi possibile integrare il monitoraggio utilizzando servizi come Amazon CloudWatch e AWS CloudTrail. Per ulteriori informazioni, consulta [the section called “Monitoraggio con CloudWatch”](#) e [the section called “Utilizzo CloudTrail con Amazon Location”](#).

## Concetti di geocodifica

Un indice dei luoghi di Amazon Location fornisce un'azione chiamata [SearchPlaceIndexForText](#) che consente di specificare il testo da cercare. Ad esempio, puoi cercare:

- Luoghi: una ricerca per **Paris** potrebbe restituire l'ubicazione della città in Francia.
- Aziende: una ricerca di **coffee shop** potrebbe restituire un elenco di caffetterie, compresi i loro nomi e sedi. Puoi anche specificare una località in cui effettuare la ricerca o un riquadro di selezione in cui effettuare la ricerca, per rendere i risultati più pertinenti. In questo caso, fornire una sede nel centro di Seattle, Washington, restituirebbe le caffetterie di quella zona.
- Indirizzi: una ricerca **1600 Pennsylvania Ave, Washington D.C.** potrebbe restituire l'ubicazione della Casa Bianca negli Stati Uniti (che si trova a quell'indirizzo).

La ricerca di testo in questo modo viene generalmente definita geocodifica, che consiste nel trovare una posizione geografica per l'indirizzo o il luogo.

Amazon Location Service fornisce anche un'azione di geocodifica inversa chiamata.

[SearchPlaceIndexForPosition](#) Questa operazione prende una posizione geografica e restituisce l'indirizzo, l'azienda o altre informazioni su ciò che si trova in quella località.

## Risultati della ricerca

Quando effettui una richiesta di ricerca con successo in Amazon Location Service, vengono restituiti uno o più risultati. Ogni risultato include un'etichetta, che è il nome o la descrizione del risultato. Ad esempio, una ricerca di **coffee shop**, potrebbe restituire un risultato con l'etichetta **Hometown Cafe** che indica che è stata trovata una caffetteria chiamata «Hometown Cafe». Il risultato della ricerca include in genere anche un indirizzo strutturato (con proprietà quali numero di indirizzo, unità, via e codice postale). A seconda del fornitore di dati, includerà anche altri metadati, come il paese e il fuso orario.

Per una ricerca in base al nome o alla categoria di un'azienda (ad esempio **coffee shop**), potresti voler mostrare tutti i risultati restituiti su una mappa. Per una ricerca di indirizzi, potresti voler utilizzare automaticamente il primo risultato. Per informazioni sulla pertinenza, consulta l'argomento successivo.

## Molteplici risultati e rilevanza

Quando si esegue una ricerca per testo, Amazon Location Service trova spesso più di un risultato. Ad esempio, una ricerca di **Paris** può restituire la città in Francia, ma anche la città del Texas. I risultati vengono ordinati in base alla pertinenza, in base a quanto stabilito dal fornitore di dati.

### Note

I risultati vengono restituiti in ordine di rilevanza da tutti i fornitori. Se si sceglie Esri o Grab come fornitore di dati, i risultati includono un valore di pertinenza che è possibile utilizzare per comprendere la pertinenza relativa tra i risultati di una singola richiesta.

La specificazione di informazioni aggiuntive, come il nome di un paese o una località in cui effettuare la ricerca, può modificare l'ordine dei risultati, ridurre il numero di risultati o persino modificare il set di risultati restituiti. Ad esempio, è più probabile che una ricerca **Paris** con una località in Texas in cui effettuare la ricerca venga restituita **Paris, Texas** come primo risultato rispetto a **Paris, France**.

In un'applicazione interattiva, è possibile utilizzare la pertinenza per decidere se accettare il risultato principale o per chiedere a un utente di chiarire i diversi risultati restituiti. Se il primo risultato ha un'elevata rilevanza, potresti semplicemente accettarlo come risposta corretta. Se sono presenti più risultati ad alta pertinenza o nessun risultato di elevata rilevanza, potresti voler elencare i risultati e consentire all'utente di selezionare il risultato migliore.

## Indirizza i risultati

Puoi cercare indirizzi con Amazon Location Service utilizzando la stessa [SearchPlaceIndexForText](#) azione. Più informazioni fornisci, più è probabile che l'indirizzo restituito corrisponda a quello fornito. Ad esempio, **123 Main St** è meno probabile che trovi un risultato corretto rispetto a **123 Main St, Anytown, California, 90210**.

Gli indirizzi hanno più attributi, come il numero civico, la via, la città, la regione e il codice postale, ecc. Questi attributi vengono utilizzati per trovare un indirizzo nell'indice dei luoghi che corrisponda al maggior numero possibile di aspetti. Maggiore è il numero di attributi trovati, più pertinente viene considerata la corrispondenza e maggiore è la probabilità che venga restituita.

### Note

La rilevanza per i risultati degli indirizzi si basa sulla corrispondenza tra il risultato e l'input. Questo potrebbe essere il numero degli attributi corrispondenti, ma anche la corrispondenza tra i risultati e l'input. Ad esempio, un input di **123 Main St** avrebbe una rilevanza maggiore quando **Main St** viene trovato nei dati, rispetto a **Main St** come unico risultato. **Main St** verrà comunque restituito, ma probabilmente con un valore di pertinenza inferiore.

I risultati della ricerca includono un'etichetta per l'indirizzo completo (123 Main St, Anytown, California, 90210), ma anche i singoli attributi strutturati dell'indirizzo restituito. Ciò è utile perché è possibile utilizzarlo, ad esempio, per compilare i campi degli indirizzi in un database o per esaminare i risultati e trovare la città, la regione o il codice postale della località trovata.

## Interpolazione

I dati dell'indice degli indirizzi includono le corrispondenze esatte degli indirizzi. Ad esempio, supponiamo che ci sia una strada 9th street e che un isolato abbia 2 case 220 e 240, come nell'immagine seguente.



Il fornitore di dati crea i dati di geolocalizzazione con questi due indirizzi noti. Puoi cercare questi due indirizzi e verranno trovati. Dopo che il fornitore di dati ha creato i dati della mappa, supponiamo che venga aggiunta una nuova casa tra i primi due indirizzi. A questa nuova casa viene dato l'indirizzo **230**. Se cerchi **230 S 9th St**, il fornitore di dati troverà comunque un risultato. Invece di utilizzare un indirizzo noto, eseguirà l'interpolazione tra gli indirizzi già noti e stimerà la posizione del nuovo indirizzo in base a questi. In questo caso, si potrebbe supporre che 230 sia a metà strada tra 220 e 240 (e sullo stesso lato della strada) e restituire una posizione approssimativa in base a tale posizione.

#### Note

I fornitori di dati aggiornano periodicamente i propri dati di geolocalizzazione con nuovi indirizzi. In questo caso, **230 S 9th St** verrebbero aggiunti ai dati del fornitore di dati, ma in genere c'è un periodo in cui un nuovo indirizzo è stato creato ma non è ancora stato aggiunto ai dati.

In questo caso, il fornitore di dati non è in grado di stabilire se il nuovo indirizzo esiste nel mondo, in quanto non è ancora presente nei dati, ma fornisce la risposta migliore possibile sulla base delle informazioni in suo possesso. Questo risultato viene chiamato interpolato e può essere restituito dal fornitore di dati nei risultati. Se `interpolated` restituisce `false`, si tratta di un indirizzo noto.

Se restituisci `true`, è un indirizzo approssimativo. Se non viene restituito, il fornitore di dati non ha fornito le informazioni sul fatto che il risultato provenga dall'interpolazione.

### Important

Il fornitore di dati può anche restituire risultati interpolati per indirizzi che non esistono affatto. Ad esempio, in questo caso, se hai inserito **232 S 9th St**, il provider troverà questo indirizzo inesistente e restituirà una posizione vicina a 230, ma sul lato 240. Gli indirizzi interpolati sono utili per raggiungere la posizione corretta, ma è bene tenere presente che non sono indirizzi noti.

## Memorizzazione dei risultati della geocodifica

Quando si crea una risorsa relativa all'indice dei luoghi, è necessario specificare un'opzione di archiviazione dei dati (chiamata `IntendedUse` nell'API). Possono essere impostati come risultati monouso o memorizzati. Si tratta di una domanda sull'uso previsto dei risultati. Se intendi archiviare i risultati (anche per scopi di memorizzazione nella cache), devi scegliere l'opzione di archiviazione, non l'opzione monouso.

### Note

Quando scegli l'opzione memorizzata (etichettata come Sì, i risultati verranno archiviati nella console o scelti storage nell'`CreatePlaceIndexAPI`), Amazon Location Service non memorizza i risultati per te. Ciò indica che intendi archiviare i risultati.

Quando esamini come utilizzerai i risultati delle tue richieste ad Amazon Location Service, devi sempre conoscere i [Termini di AWS servizio](#) applicabili.

## Terminologia relativa ai luoghi

### Risorsa dell'indice dei luoghi

Consente di scegliere una fonte di dati per supportare le query di ricerca. Ad esempio, puoi cercare punti di interesse, indirizzi o coordinate. Quando una query di ricerca viene inviata a una risorsa dell'indice dei luoghi, viene soddisfatta utilizzando l'origine dati configurata della risorsa.

## Geocodifica

La geocodifica è il processo che prevede l'immissione di un testo, la ricerca nell'indice dei luoghi e la restituzione dei risultati con la posizione.

## Geocodifica inversa

La geocodifica inversa è il processo di acquisizione di una posizione e restituzione di informazioni su tale posizione dall'indice dei luoghi, come l'indirizzo, la città o l'attività commerciale in quella posizione.

## Relevance

La pertinenza è la corrispondenza tra un risultato e l'input. Non è una misura di correttezza.

## Interpolazione

L'interpolazione è il processo di ricerca di indirizzi sconosciuti utilizzando posizioni di indirizzi note come punti guida.

## Codici dei paesi ISO 3166

Amazon Location Service Places utilizza [i codici paese dell'International Organization for Standardization \(ISO\) 3166](#) per fare riferimento a paesi o regioni.

Per trovare il codice per un paese o una regione specifici, usa la piattaforma di navigazione [online ISO](#).

## Route

Questa sezione fornisce una panoramica dei concetti relativi al routing tramite Amazon Location Service.

### Note

Per informazioni su come utilizzare in pratica le risorse di routing, consulta [Calcolo dei percorsi utilizzando Amazon Location Service](#).

## Risorse per il calcolatore del percorso

Le risorse per il calcolo dei percorsi ti consentono di trovare percorsi e stimare il tempo di viaggio in base alla rete up-to-date stradale e alle informazioni sul traffico in tempo reale fornite dal fornitore di dati scelto.

Puoi utilizzare le API Routes per creare funzionalità che consentano all'applicazione di richiedere il tempo di viaggio, la distanza e la geometria del percorso tra due località qualsiasi. Puoi anche utilizzare l'API Routes per richiedere il tempo di viaggio e la distanza tra un insieme di partenze e destinazioni in un'unica richiesta per calcolare una matrice.

Di seguito viene illustrato come creare e utilizzare una risorsa per il calcolo del percorso:



1. Innanzitutto, crei una risorsa per il calcolo del percorso nel tuo AWS account selezionando un fornitore di dati.
2. Puoi quindi selezionare e installare l'SDK più adatto al tuo ambiente di sviluppo e alle tue applicazioni.
3. Inizia a utilizzare le API Amazon Location Routes. Per ulteriori informazioni su come utilizzare le API di routing, consulta l'argomento on. [Calcolo dei percorsi utilizzando Amazon Location Service](#)
4. È quindi possibile integrare il monitoraggio utilizzando servizi come Amazon CloudWatch e AWS CloudTrail. Per ulteriori informazioni, consulta [Monitoraggio di Amazon Location Service con Amazon CloudWatch](#) e [Registrazione e monitoraggio con AWS CloudTrail](#).

## Calcolo di un percorso

Una risorsa per il calcolo del percorso di Amazon Location fornisce un'azione chiamata `CalculateRoute` che puoi utilizzare per creare un percorso tra due località geografiche (la partenza e la destinazione). Il percorso calcolato include la geometria per disegnare il percorso su una mappa, più il tempo e la distanza complessivi del percorso.

### Utilizzo dei waypoint



Quando crei la tua richiesta di percorso, puoi aggiungere ulteriori waypoint al percorso. Si tratta di punti tra la partenza e la destinazione che fungono da fermate lungo il percorso. Il percorso verrà calcolato attraverso ciascuno dei waypoint specificati. Il percorso da un punto della richiesta al successivo si chiama aLeg. Ogni tappa include una distanza, un tempo e la geometria per quella parte del percorso.

#### Note

I waypoint vengono instradati nell'ordine indicato nella richiesta. Non vengono riordinati per il percorso più breve. Consulta la [Pianificazione dei percorsi](#) sezione per informazioni su come trovare il percorso più breve.

Puoi includere fino a 25 waypoint in una singola richiesta per calcolare un percorso.

#### Traffico e orario di partenza

Amazon Location Service tiene conto del traffico nel calcolo di un percorso. Il traffico che considera si basa sull'ora specificata. Puoi specificare di partire adesso oppure puoi fornire un orario specifico in cui desideri partire, che influirà sul risultato del percorso adattandolo al traffico all'ora specificata.

#### Note

È possibile calcolare l'orario di arrivo utilizzando l'orario di partenza e il tempo di risposta del percorso, ad esempio per stimare l'arrivo di un conducente.

Se desideri che Amazon Location non tenga conto del traffico, non specificare l'orario di partenza e non specificare la partenza ora. Questo calcolerà un percorso che presuppone le migliori condizioni di traffico per il percorso.

#### Opzioni della modalità di viaggio

Puoi impostare la modalità di viaggio durante il calcolo di un percorso utilizzando Amazon Location Service. La modalità di viaggio predefinita è auto, ma puoi alternativamente selezionare camion o camminare.

Se si specifica la modalità auto o camion, è possibile specificare anche opzioni aggiuntive.

Per la modalità auto, puoi specificare che desideri evitare le strade a pedaggio o i traghetti. In questo modo si cercherà di evitare traghetti e strade a pedaggio, ma si continuerà a percorrerli, se sono l'unico modo per raggiungere la destinazione.

In modalità camion, puoi anche evitare traghetti e strade a pedaggio, ma puoi anche specificare le dimensioni e il peso del camion, per evitare percorsi che non consentano il trasporto del camion.

## Pianificazione dei percorsi

Puoi utilizzare Amazon Location Service per creare input per il tuo software di pianificazione e ottimizzazione dei percorsi. Puoi creare risultati di percorso, inclusi tempo di viaggio e distanza da percorrere, per percorsi tra una serie di posizioni di partenza e una serie di posizioni di destinazione. Questa operazione si chiama creazione di una matrice di itinerari.

### Note

Esistono molti scenari diversi che il software di pianificazione e ottimizzazione dei percorsi può risolvere. Ad esempio, il software di pianificazione può utilizzare l'insieme di tempi e distanze tra i punti per calcolare il percorso più breve che si ferma in ogni punto, fornendo un percorso efficiente per un singolo conducente. In alternativa, è possibile utilizzare un software di pianificazione per suddividere le fermate tra più camion, garantendo l'efficienza dell'intera flotta o per garantire che ogni cliente venga visitato entro il periodo di tempo richiesto. Amazon Location fornisce le funzioni di routing in modo efficiente per consentire al software di pianificazione di completare la propria attività.

Ad esempio, date le posizioni di partenza A e B e le posizioni di destinazione X e Y, Amazon Location Service restituirà il tempo di viaggio e la distanza percorsa per i percorsi da A a X, da A a Y, da B a X e da B a Y.

Come per il calcolo di un singolo itinerario, puoi calcolare i percorsi con diverse modalità di trasporto, evitamenti e condizioni del traffico. Ad esempio, è possibile specificare che il veicolo è un camion lungo 35 piedi e il percorso calcolato utilizzerà tali restrizioni per determinare il tempo di viaggio e la distanza da percorrere. Non è possibile includere i waypoint nel calcolo della matrice di percorso.

Il numero di risultati restituiti (e percorsi calcolati) è il numero di posizioni di partenza moltiplicato per il numero di posizioni di destinazione. Ti viene addebitato ogni itinerario calcolato, non ogni richiesta al servizio, quindi una matrice di itinerari con 10 partenze e 10 destinazioni verrà fatturata come 100 rotte.

## Terminologia del percorso

### Risorsa per il calcolatore del percorso

Una AWS risorsa che consente di stimare il tempo di viaggio, la distanza e tracciare i percorsi su una mappa con dati sul traffico e sulla rete stradale provenienti dal fornitore di dati scelto.

Utilizzando le risorse del calcolatore di percorso, puoi calcolare i percorsi per diverse modalità di trasporto, deviazioni e condizioni del traffico.

### Route

Un itinerario contiene i dettagli utilizzati quando si percorre un percorso dalla posizione di partenza, dalle posizioni dei waypoint e dalla posizione di destinazione.

Alcuni esempi di dettagli di un percorso includono:

- La distanza da una posizione a un'altra.
- Il tempo necessario per spostarsi da una posizione alla posizione successiva.
- La LineString geometria che rappresenta il percorso del percorso.

Per ulteriori informazioni sulle rotte, consulta la [sintassi di risposta per l' CalculateRoute operazione](#) nel riferimento all'API Amazon Location Service Routes.

### Matrice delle rotte

Un elenco di percorsi, da un insieme di posizioni di partenza a un insieme di posizioni di destinazione. Utile come input nel software di pianificazione o ottimizzazione del percorso.

Per ulteriori informazioni sul calcolo di una matrice di route, consulta la [sintassi per l' CalculateRouteMatrix operazione](#) nel riferimento all'API Amazon Location Service Routes.

### LineStringgeometria

Un itinerario Amazon Location è costituito da una o più tappe (un percorso da un waypoint all'altro all'interno del percorso complessivo). La geometria di ogni gamba è una polilinea rappresentata da un. LineString A LineString è una serie ordinata di posizioni che può essere utilizzata per tracciare un percorso su una mappa.

Quello che segue è un esempio di a LineString con tre punti:

```
[  
  [-122.7565, 49.0021],  
  [-122.3394, 47.6159],
```

```
[-122.1082, 45.8371]
```

```
]
```

## Waypoint

I waypoint sono posizioni intermedie che fungono da fermate lungo un percorso tra la posizione di partenza e la posizione di destinazione. L'ordine degli scali sul percorso segue l'ordine in cui vengono fornite le posizioni dei waypoint nella richiesta.

## Gamba

Una singola gamba è il viaggio da una posizione all'altra. Se le posizioni non si trovano su una strada, vengono spostate sulla strada più vicina. Il numero di tappe di un percorso è inferiore di uno rispetto al numero totale di posizioni.

Un itinerario senza waypoint consiste in un'unica tappa, dalla posizione di partenza alla destinazione. Un itinerario con 1 waypoint è composto da 2 tappe, dalla posizione di partenza al waypoint e poi dal waypoint alla destinazione.

## Fase

Un passo è una sottosezione di una gamba. Ogni fase fornisce informazioni riassuntive relative a quella fase della gamba.

# Geofence e tracker

Questa sezione fornisce una panoramica dei concetti relativi all'utilizzo dei geofence e dei tracker di Amazon Location Service. I geofence sono confini poligonali che puoi usare per ricevere notifiche quando dispositivi o posizioni entrano ed escono dalle aree. Le risorse Tracker vengono utilizzate per memorizzare e aggiornare le posizioni dei dispositivi mentre si spostano.

### Note

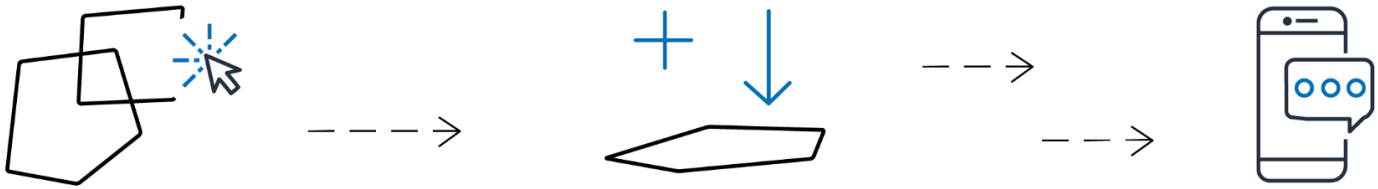
Per informazioni su come utilizzare in pratica i geofence e i tracker, consulta. [Geofencing di un'area di interesse utilizzando Amazon Location](#)

## Geofence

Le risorse per la raccolta di geofence consentono di archiviare e gestire i geofence, ovvero confini virtuali su una mappa. È possibile valutare le posizioni rispetto a una risorsa di raccolta di geofence

e ricevere notifiche quando l'aggiornamento della posizione supera il confine di uno qualsiasi dei geofence della raccolta di geofence.

Di seguito viene illustrato come creare e utilizzare le risorse di raccolta di geofence:



1. Crea una risorsa per la raccolta di geofence nel tuo account. AWS
2. Aggiungi geofence a quella raccolta. Puoi farlo utilizzando lo strumento di caricamento geofence sulla console Amazon Location o utilizzando l'API Amazon Location Geofences. Per ulteriori informazioni sulle opzioni disponibili, consulta [Accedere ad Amazon Location](#).

I geofence possono essere definiti da un poligono o da un cerchio. Usa un poligono per scoprire quando un dispositivo entra in un'area specifica. Usa un cerchio per scoprire quando un dispositivo si trova entro una certa distanza (raggio) da un punto.

3. Puoi iniziare a valutare le posizioni rispetto a tutti i tuoi geofence. Quando un aggiornamento della posizione supera i confini di uno o più geofence, la tua risorsa di raccolta di geofence emette uno dei seguenti tipi di eventi di geofence su Amazon: EventBridge
  - INVIO: viene generato un evento per ogni geofence in cui l'aggiornamento della posizione ne attraversa il confine entrando.
  - USCITA: viene generato un evento per ogni geofence in cui l'aggiornamento della posizione ne attraversa il confine uscendo.

Per ulteriori informazioni, consulta [the section called “Reagire agli eventi con EventBridge”](#). Puoi anche integrare il monitoraggio utilizzando servizi come Amazon CloudWatch e AWS CloudTrail. Per ulteriori informazioni, consulta [the section called “Monitoraggio con CloudWatch”](#) e [the section called “Utilizzo CloudTrail con Amazon Location”](#).

Ad esempio, se stai monitorando una flotta di camion e desideri ricevere una notifica quando un camion entra in una determinata area di uno dei tuoi magazzini. Puoi creare un geofence per l'area intorno a ciascun magazzino. Quindi, quando i camion ti inviano posizioni aggiornate, puoi utilizzare Amazon Location Service per valutare tali posizioni e vedere se un camion è entrato (o uscito) da una delle aree di geofence.

### Note

Ti viene fatturato in base al numero di raccolte di geofence rispetto alle quali effettui la valutazione. La fattura non è influenzata dal numero di geofence presenti in ciascuna collezione. Poiché ogni raccolta di geofence può contenere fino a 50.000 geofence, potresti voler combinare i geofence in un numero inferiore di raccolte, ove possibile, per ridurre i costi delle valutazioni dei geofence. Gli eventi generati includeranno l'ID del singolo geofence della raccolta, nonché l'ID della raccolta.

## Eventi Geofence

Le posizioni che stai monitorando sono referenziate da un ID chiamato a `DeviceId` (e le posizioni sono chiamate posizioni dei dispositivi). Puoi inviare un elenco di posizioni dei dispositivi da valutare direttamente alla risorsa di raccolta geofence oppure puoi utilizzare un tracker. Consulta la sezione successiva per ulteriori informazioni sui tracker.

Ricevi eventi (tramite Amazon EventBridge) solo quando un dispositivo entra o esce da un geofence, non per ogni cambio di posizione. Ciò significa che in genere riceverai eventi e dovrai rispondere ad essi molto meno frequentemente rispetto a ogni aggiornamento della posizione del dispositivo.

### Note

Per la prima valutazione della posizione di un determinato dispositivo `DeviceID`, si presume che in precedenza il dispositivo non si trovasse in alcun geofence. Quindi il primo aggiornamento genererà un `ENTER` evento, se all'interno di un geofence della raccolta, e nessun evento in caso contrario.

Per calcolare se un dispositivo è entrato o uscito da un geofence, Amazon Location Service deve mantenere lo stato della posizione precedente del dispositivo. Questo stato di posizione viene memorizzato per 30 giorni. Dopo 30 giorni senza un aggiornamento per un dispositivo, un nuovo aggiornamento della posizione verrà considerato come il primo aggiornamento della posizione.

## Localizzatori

Un tracker memorizza gli aggiornamenti di posizione per una raccolta di dispositivi. Il tracker può essere utilizzato per interrogare la posizione corrente o la cronologia delle posizioni dei dispositivi.

Memorizza gli aggiornamenti, ma riduce lo spazio di archiviazione e il rumore visivo filtrando le posizioni prima di archivarle.

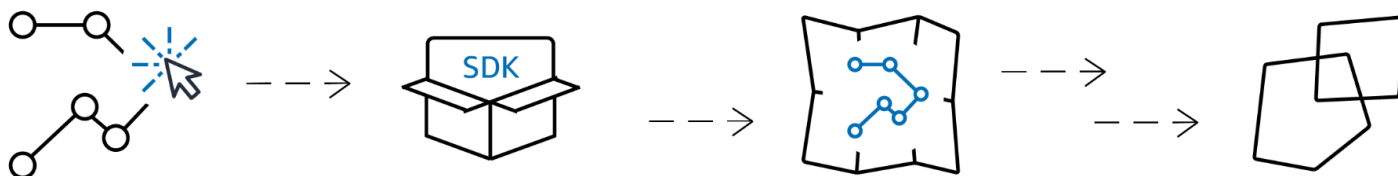
Ogni aggiornamento di posizione memorizzato nelle risorse del tracker può includere una misura della precisione della posizione e fino a 3 campi di metadati sulla posizione o sul dispositivo che desideri memorizzare. I metadati vengono archiviati come coppie chiave-valore e possono memorizzare informazioni come velocità, direzione, pressione dei pneumatici o temperatura del motore.

### Note

L'archiviazione dei tracker viene crittografata automaticamente con chiavi di proprietà. AWS Puoi aggiungere un altro livello di crittografia utilizzando le chiavi KMS che gestisci, per garantire che solo tu possa accedere ai tuoi dati. Per ulteriori informazioni, consulta [Crittografia dei dati a riposo per Amazon Location Service](#).

Il filtraggio e l'archiviazione della posizione dei tracker sono utili di per sé, ma i tracker sono particolarmente utili se abbinati a geofence. Puoi collegare i tracker a una o più delle tue risorse di raccolta di geofence e gli aggiornamenti di posizione vengono valutati automaticamente rispetto ai geofence presenti in tali raccolte. Un uso corretto dei filtri può inoltre ridurre notevolmente i costi delle valutazioni dei geofence.

Il diagramma seguente mostra come creare e utilizzare le risorse del tracker:



1. Innanzitutto, crei una risorsa tracker nel tuo account. AWS
2. Quindi, decidi come inviare gli aggiornamenti sulla posizione alle risorse del tracker. Usa [AWS gli SDK](#) per integrare le funzionalità di tracciamento nelle tue applicazioni mobili. In alternativa, puoi utilizzare MQTT seguendo le step-by-step istruzioni per il [tracciamento](#) tramite MQTT.
3. Ora puoi utilizzare la tua risorsa tracker per registrare la cronologia delle posizioni e visualizzarla su una mappa.
4. Puoi anche collegare la tua risorsa di localizzazione a una o più raccolte di geofence in modo che ogni aggiornamento di posizione inviato alla risorsa tracker venga automaticamente valutato

rispetto a tutti i geofence di tutte le raccolte di geofence collegate. Puoi collegare la risorsa nella pagina dei dettagli delle risorse tracker della console Amazon Location o utilizzando l'API Amazon Location Trackers.

5. È quindi possibile integrare il monitoraggio utilizzando servizi come Amazon CloudWatch e AWS CloudTrail. Per ulteriori informazioni, consulta [the section called “Monitoraggio con CloudWatch”](#) e [the section called “Utilizzo CloudTrail con Amazon Location”](#).

## Utilizzo di tracker con geofence

I tracker forniscono funzionalità aggiuntive se abbinati a geofence. Associate un tracker a una raccolta di geofence, tramite la console Amazon Location o l'API, per valutare automaticamente le posizioni dei tracker. Ogni volta che il tracker riceve una posizione aggiornata, tale posizione verrà valutata rispetto a ogni geofence della raccolta e gli EXIT eventi ENTER appropriati verranno generati in Amazon. EventBridge Puoi anche applicare filtri al tracker e, a seconda del filtro, ridurre i costi per le valutazioni del geofence valutando solo gli aggiornamenti significativi della posizione.

Se si associa il tracker a una raccolta di geofence dopo che quest'ultima ha già ricevuto alcuni aggiornamenti di posizione, il primo aggiornamento di posizione dopo l'associazione viene considerato come un aggiornamento iniziale per le valutazioni del geofence. Se si trova all'interno di un geofence, riceverai un evento. ENTER Se non si trova all'interno di alcun geofence, non riceverai alcun EXIT evento, indipendentemente dallo stato precedente.

## Filtraggio della posizione

I tracker possono filtrare automaticamente le posizioni che vengono loro inviate. Esistono diversi motivi per cui potresti voler filtrare alcuni aggiornamenti sulla posizione del tuo dispositivo. Se disponi di un sistema che invia report solo ogni minuto circa, potresti voler filtrare i dispositivi in base all'ora, archiviando e valutando le posizioni solo ogni 30 secondi. Anche se effettui il monitoraggio più frequentemente, potresti voler filtrare gli aggiornamenti di posizione per eliminare la rumorosità dell'hardware GPS. Le posizioni GPS sono intrinsecamente rumorose. La loro precisione non è perfetta al 100%, quindi anche un dispositivo fermo sembra muoversi leggermente. A basse velocità, questo jitter causa confusione visiva e può causare falsi eventi di entrata e uscita se il dispositivo si trova vicino al bordo di un geofence.

Il filtro di posizione funziona come se un tracker ricevesse gli aggiornamenti di posizione, riducendo il rumore visivo nei percorsi del dispositivo (jitter), il numero di falsi eventi di entrata e uscita dal geofence e aiutando a gestire i costi riducendo il numero di aggiornamenti di posizione archiviati e di valutazioni del geofence attivate.



I tracker offrono tre opzioni di filtraggio della posizione per aiutare a gestire i costi e ridurre il jitter negli aggiornamenti della posizione.

- **Basato sulla precisione:** utilizzabile con qualsiasi dispositivo che fornisce una misurazione di precisione. La maggior parte dei dispositivi GPS e mobili fornisce queste informazioni. La precisione di ogni misurazione della posizione è influenzata da molti fattori ambientali, tra cui la ricezione satellitare GPS, il paesaggio e la vicinanza dei dispositivi wifi e bluetooth. La maggior parte dei dispositivi, inclusa la maggior parte dei dispositivi mobili, è in grado di fornire una stima dell'accuratezza della misurazione insieme alla misurazione. Con AccuracyBased il filtraggio, Amazon Location ignora gli aggiornamenti sulla posizione se il dispositivo si è spostato meno della precisione misurata. Ad esempio, se due aggiornamenti consecutivi da un dispositivo hanno un intervallo di precisione compreso tra 5 m e 10 m, Amazon Location ignora il secondo aggiornamento se il dispositivo si è spostato a meno di 15 m. Amazon Location non valuta gli aggiornamenti ignorati rispetto ai geofence, né li archivia.

Se la precisione non viene fornita, viene considerata pari a zero e la misurazione viene considerata perfettamente accurata e non verrà applicato alcun filtro agli aggiornamenti.

#### Note

È possibile utilizzare filtri basati sulla precisione per rimuovere tutti i filtri. Se selezioni il filtraggio basato sulla precisione, ma sostituisci tutti i dati di precisione a zero o ometti completamente la precisione, Amazon Location non filtrerà alcun aggiornamento.

Nella maggior parte degli scenari, il filtraggio basato sulla precisione è una buona scelta per filtrare gli aggiornamenti di posizione, in quanto consente di tracciare la posizione ed escludere gli aggiornamenti non necessari, riducendo così i costi.

- **Basato sulla distanza:** da utilizzare quando i dispositivi non forniscono una misurazione di precisione, ma si desidera comunque sfruttare il filtraggio per ridurre il jitter e gestire i costi. DistanceBased il filtraggio ignora gli aggiornamenti sulla posizione in cui i dispositivi si sono spostati a meno di 30 m (98,4 piedi). Quando utilizzi il filtro di DistanceBased posizione, Amazon Location non valuta questi aggiornamenti ignorati rispetto ai geofence né archivia gli aggiornamenti.

La precisione della maggior parte dei dispositivi mobili, inclusa la precisione media dei dispositivi iOS e Android, è entro 15 m. Nella maggior parte delle applicazioni, il DistanceBased filtraggio può ridurre l'effetto delle imprecisioni di posizione quando si visualizza la traiettoria del dispositivo

su una mappa e l'effetto di rimbalzo di più eventi consecutivi di entrata e uscita quando i dispositivi si trovano vicino al confine di un geofence. Può anche contribuire a ridurre il costo dell'applicazione, effettuando meno chiamate per effettuare valutazioni sulla base di geofence collegati o recuperare le posizioni dei dispositivi.

Il filtro basato sulla distanza è utile se desideri filtrare, ma il tuo dispositivo non fornisce misurazioni di precisione o desideri filtrare un numero maggiore di aggiornamenti rispetto a quello basato sulla precisione.

- **Basato sul tempo:** (impostazione predefinita) Utilizzalo quando i dispositivi inviano aggiornamenti di posizione molto frequentemente (più di una volta ogni 30 secondi) e desideri ottenere valutazioni del geofence quasi in tempo reale senza archiviare ogni aggiornamento. Nel TimeBased filtraggio, ogni aggiornamento di posizione viene valutato sulla base di raccolte di geofence collegate, ma non tutti gli aggiornamenti di posizione vengono archiviati. Se la frequenza di aggiornamento è superiore a 30 secondi, viene archiviato un solo aggiornamento ogni 30 secondi per ogni ID dispositivo univoco.

Il filtraggio basato sul tempo è particolarmente utile quando si desidera memorizzare meno posizioni, ma si desidera che ogni aggiornamento di posizione venga valutato rispetto alle raccolte di geofence associate.

#### Note

Fai attenzione ai costi della tua applicazione di tracciamento quando decidi il metodo di filtraggio e la frequenza degli aggiornamenti delle posizioni. Ti verrà addebitato ogni aggiornamento della posizione e una volta per la valutazione dell'aggiornamento della posizione rispetto a ciascuna raccolta di geofence collegata. Ad esempio, quando si utilizza il filtro basato sul tempo, se il tracker è collegato a due raccolte di geofence, ogni aggiornamento di posizione verrà conteggiato come una richiesta di aggiornamento della posizione e due valutazioni della raccolta di geofence. Se segnali aggiornamenti di posizione ogni 5 secondi per i tuoi dispositivi e utilizzi il filtro basato sul tempo, ti verranno fatturati 720 aggiornamenti della posizione e 1.440 valutazioni di geofence all'ora per ciascun dispositivo.

# Terminologia Geofence

## Collezione Geofence

Contiene zero o più geofence. È in grado di monitorare i geofence emettendo eventi di entrata e uscita, quando richiesto, per valutare la posizione di un dispositivo rispetto ai relativi geofence.

## Geofence

Una geometria poligonale o circolare che definisce un confine virtuale su una mappa.

## Geometria poligonale

Un geofence Amazon Location è un confine virtuale per un'area geografica ed è rappresentato come una geometria poligonale o come un cerchio.

Un cerchio è un punto con una distanza attorno ad esso. Usa un cerchio per ricevere una notifica se un dispositivo si trova entro una certa distanza da una posizione.

Un poligono è una matrice composta da uno o più anelli lineari. Utilizzate un poligono quando desiderate definire un limite specifico per le notifiche del dispositivo. Un anello lineare è una matrice di quattro o più vertici, in cui il primo e l'ultimo vertice coincidono per formare un limite chiuso. Ogni vertice è un punto bidimensionale della forma *[longitudine, latitudine]*, dove le unità di *longitudine e latitudine* sono gradi. I vertici devono essere elencati in senso antiorario attorno al poligono.

### Note

Amazon Location Service non supporta poligoni con più di un anello. Ciò include fori, isole o multipoligoni. Amazon Location inoltre non supporta poligoni avvolti in senso orario o che attraversano l'antimeridiano.

Di seguito è riportato un esempio di un singolo anello lineare esterno:

```
[  
  [  
    [-5.716667, -15.933333],
```

```
[[-14.416667, -7.933333],  
 [-12.316667, -37.066667],  
 [-5.716667, -15.933333]  
 ]  
 ]
```

## Terminologia del tracker

### Risorsa Tracker

Una AWS risorsa che riceve aggiornamenti sulla posizione dai dispositivi. La risorsa tracker fornisce supporto per le query sulla posizione, ad esempio la posizione attuale e storica del dispositivo. Il collegamento di una risorsa tracker a una raccolta di geofence valuta automaticamente gli aggiornamenti della posizione rispetto a tutti i geofence della raccolta di geofence collegata.

### Dati di posizione tracciati

Una risorsa di tracciamento memorizza le informazioni sui tuoi dispositivi nel tempo. Le informazioni includono una serie di aggiornamenti sulla posizione, in cui ogni aggiornamento include posizione, ora e metadati opzionali. I metadati possono includere la precisione di una posizione e fino a tre coppie chiave-valore per aiutarti a tenere traccia delle informazioni chiave su ciascuna posizione, come velocità, direzione, pressione dei pneumatici, carburante residuo o temperatura del motore del veicolo che stai monitorando. I tracker conservano la cronologia delle posizioni del dispositivo per 30 giorni.

### Filtraggio della posizione

Il filtro di posizione può aiutarvi a controllare i costi e migliorare la qualità dell'applicazione di tracciamento filtrando gli aggiornamenti di posizione che non forniscono informazioni preziose prima che gli aggiornamenti vengano archiviati o valutati rispetto ai geofence.

Puoi scegliere AccuracyBased o filtrare. DistanceBased TimeBased Per impostazione predefinita, il filtro di posizione è impostato su. TimeBased

È possibile configurare il filtro di posizione quando si creano o si aggiornano le risorse del tracker.

### Formato timestamp RFC 3339

Amazon Location Service Trackers utilizza il formato [RFC 3339](#), che segue il formato dell'[International Organization for Standardization \(ISO\) 8601](#) per date e ora.

Il formato è «YYYY-MM-DDTHH:mm:ss.sssz+ 00:00»:

- YYYY-MM-DD— Rappresenta il formato della data.
- T— Indica che i valori temporali seguiranno.
- hh:mm:ss.sss— Rappresenta l'ora in formato 24 ore.
- Z— Indica che il fuso orario utilizzato è UTC, che può essere seguito da deviazioni dal fuso orario UTC.
- +00:00— Indicare facoltativamente le deviazioni dal fuso orario UTC. Ad esempio, + 01:00 indica UTC + 1 ora.

### Esempio

Per il 2 luglio 2020, alle 12:15:20 del pomeriggio, con un ulteriore adeguamento di 1 ora al fuso orario UTC.

```
2020-07-02T12:15:20.000Z+01:00
```

## Casi d'uso comuni per l'utilizzo di Amazon Location Service

Amazon Location Service ti consente di creare una gamma di applicazioni, dal monitoraggio delle risorse al marketing basato sulla posizione. I seguenti sono casi d'uso comuni:

### Coinvolgimento degli utenti e geomarketing

Utilizza i dati sulla posizione per creare soluzioni che migliorino il coinvolgimento degli utenti nel marketing mirato ai clienti. Ad esempio, Amazon Location può attivare un evento che richiede una notifica quando un cliente che ha ordinato un caffè sulla sua app mobile si trova nelle vicinanze. Inoltre, puoi creare funzionalità di targeting geografico in modo che i rivenditori possano inviare codici sconto o volantini digitali ai clienti che si trovano nelle vicinanze dei negozi target.

### Monitoraggio delle risorse

Sviluppa funzionalità di tracciamento delle risorse per aiutare le aziende a comprendere l'ubicazione attuale e storica dei loro prodotti, del personale e dell'infrastruttura. Con le funzionalità di tracciamento delle risorse, puoi creare una serie di soluzioni che ottimizzano il personale remoto, proteggono le spedizioni lungo il percorso e massimizzano l'efficacia delle spedizioni.

## Consegna

Integra le funzionalità di localizzazione nelle applicazioni di consegna per archiviare, tracciare e coordinare il luogo di partenza, i veicoli di consegna e la loro destinazione. Ad esempio, un'applicazione di consegna di cibo con funzionalità di Amazon Location integrate dispone di funzionalità di tracciamento della posizione e geofencing che possono avvisare automaticamente un ristorante quando un addetto alle consegne si trova nelle vicinanze. Ciò riduce i tempi di attesa e aiuta a mantenere la qualità del cibo consegnato.

Questo argomento fornisce una panoramica dell'architettura e dei passaggi per le applicazioni che puoi creare con Amazon Location.

### Argomenti

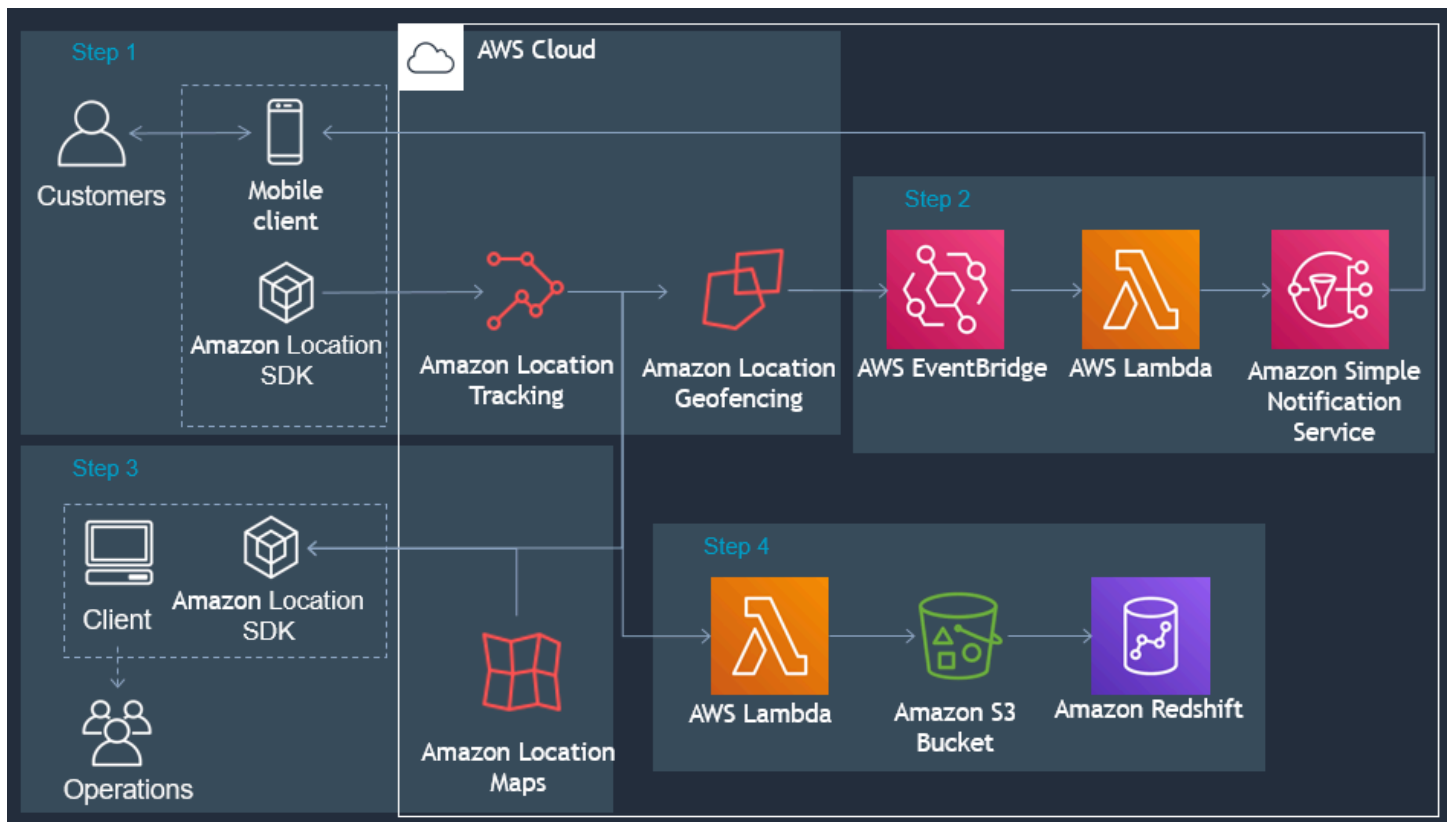
- [Applicazioni di geomarketing e coinvolgimento degli utenti](#)
- [Applicazioni di tracciamento delle risorse](#)
- [Applicazioni di distribuzione](#)

## Applicazioni di geomarketing e coinvolgimento degli utenti

Di seguito è riportata un'illustrazione di un'architettura applicativa di geomarketing e coinvolgimento degli utenti che utilizza Amazon Location:

Con questa architettura, puoi:

- Avvia eventi in base alla vicinanza di un target in modo da poter inviare offerte ai clienti vicini o coinvolgere coloro che hanno lasciato di recente la tua struttura (il cosiddetto targeting geografico).
- Visualizza le posizioni dei dispositivi dei clienti su una mappa per monitorare le tendenze nel tempo.
- Memorizza le posizioni dei dispositivi dei clienti che puoi analizzare nel tempo.
- Analizza la cronologia delle sedi per identificare tendenze e opportunità di ottimizzazione.



Di seguito è riportata una panoramica dei passaggi necessari per creare un'applicazione di geomarketing e coinvolgimento degli utenti:

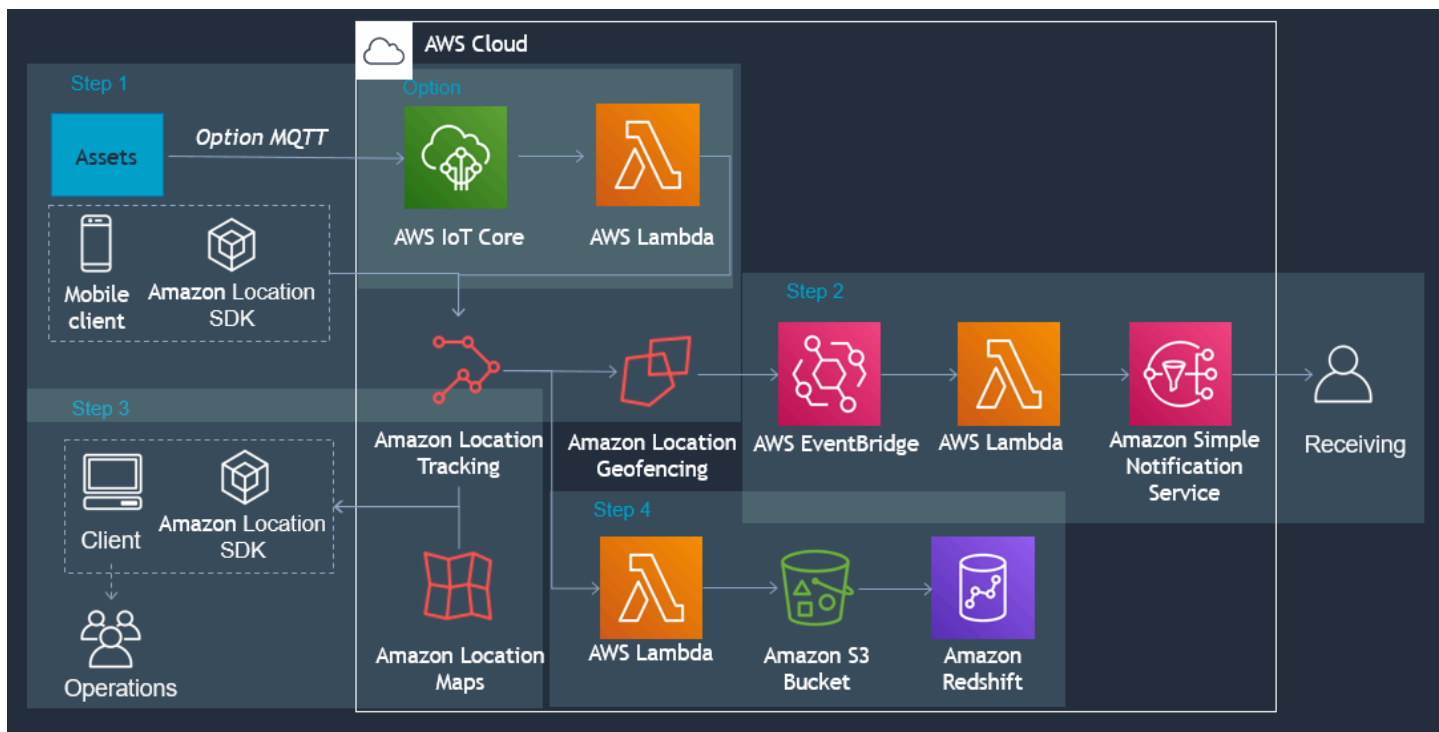
1. Crea i tuoi geofence in Geofence Collections e collega i Tracker ad essi. Per ulteriori informazioni, consulta [the section called “Geofencing e tracciamento”](#).
2. Configura Amazon EventBridge per inviare una notifica ai clienti che entrano o escono da un'area di interesse geofentata. Per ulteriori informazioni, consulta [the section called “Reagire agli eventi con EventBridge”](#).
3. Mostra le posizioni e i geofence dei clienti su una mappa. Per ulteriori informazioni, consulta [Utilizzo](#) delle mappe.
4. Salva i dati sulla posizione in una memoria a lungo termine per ulteriori analisi.
5. Dopo aver creato la tua applicazione, puoi utilizzare Amazon CloudWatch e AWS CloudTrail gestirla. Per ulteriori informazioni, consulta [the section called “Monitoraggio con CloudWatch”](#) e [the section called “Utilizzo CloudTrail con Amazon Location”](#).

## Applicazioni di tracciamento delle risorse

Di seguito è riportata un'illustrazione dell'architettura di un'applicazione di tracciamento delle risorse che utilizza Amazon Location:

Con questa architettura, puoi:

- Visualizza le posizioni delle risorse su una mappa per illustrare il quadro generale. Ad esempio, mostrando una mappa termica utilizzando luoghi o eventi storici per aiutare un team operativo o di pianificazione.
- Avvia eventi basati sulla prossimità degli asset per avvisare il reparto ricevente, in modo da prepararsi all'arrivo di una spedizione e ridurre i tempi di elaborazione.
- Memorizza le posizioni degli asset per avviare azioni nelle applicazioni di backend o per analizzare i dati nel tempo.
- Analizza la cronologia delle posizioni per identificare tendenze e opportunità di ottimizzazione.



Di seguito viene fornita una panoramica dei passaggi necessari per creare un'applicazione di tracciamento degli asset:

1. Crea i tuoi geofence in Geofence Collections e collega i Tracker ad essi. Per ulteriori informazioni, consulta [the section called "Geofencing e tracciamento"](#).



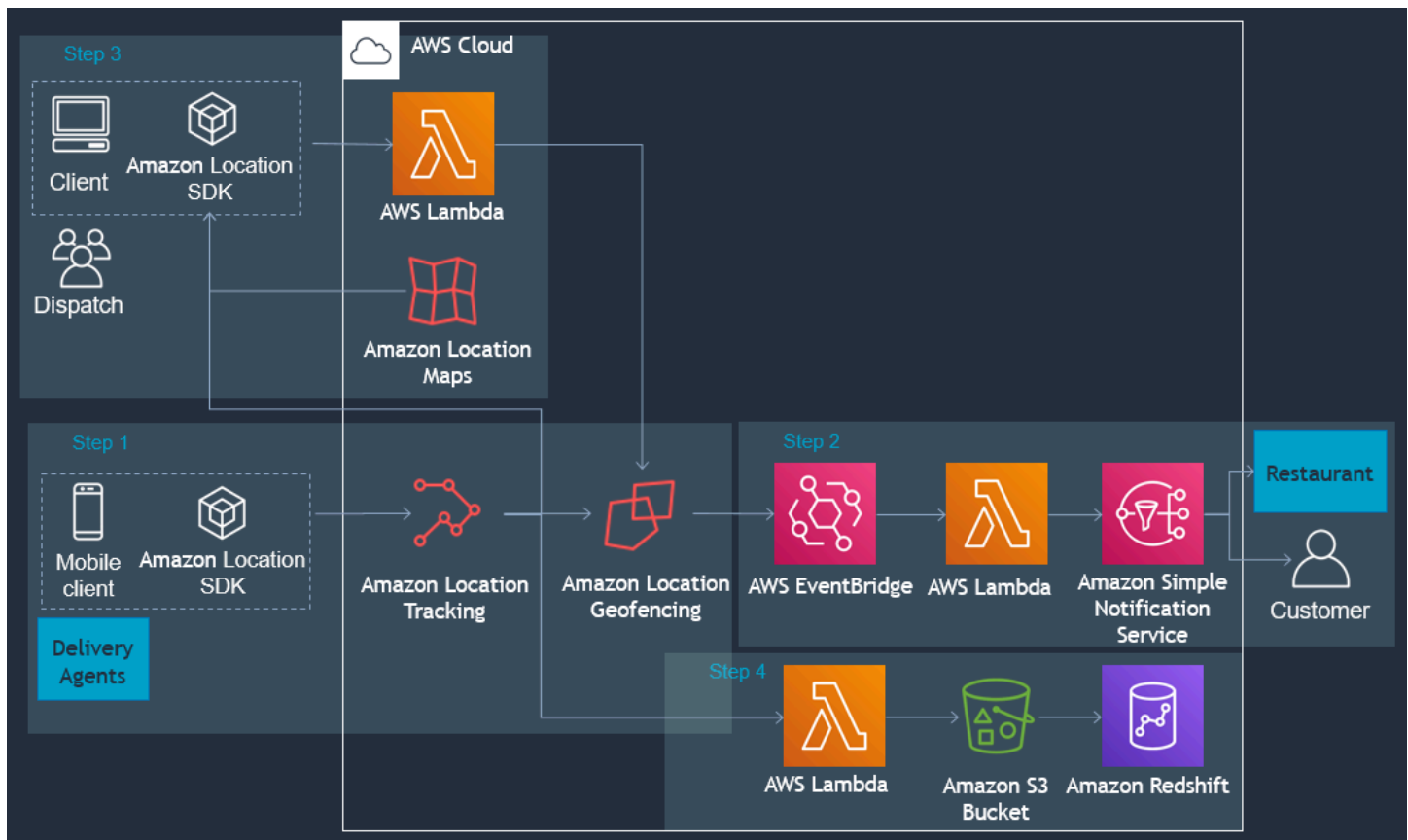
2. Configura Amazon EventBridge per inviare una notifica o avviare un processo. Per ulteriori informazioni, consulta [the section called “Reagire agli eventi con EventBridge”](#).
3. Visualizza gli asset tracciati e i geofence attivi su una mappa. [Per ulteriori informazioni, consulta Utilizzo delle mappe.](#)
4. Salva i dati sulla posizione in una memoria a lungo termine per ulteriori analisi.
5. Dopo aver creato la tua applicazione, puoi utilizzare Amazon CloudWatch e AWS CloudTrail gestirla. Per ulteriori informazioni, consulta [the section called “Monitoraggio con CloudWatch”](#) e [the section called “Utilizzo CloudTrail con Amazon Location”](#).

## Applicazioni di distribuzione

Di seguito è riportata un'illustrazione dell'architettura di un'applicazione di distribuzione che utilizza Amazon Location.

Con questa architettura, puoi:

- Avvia eventi in base alla vicinanza degli agenti di consegna in modo che il ritiro sia pronto in tempo e che i clienti possano essere avvisati dell'arrivo della consegna.
- Visualizza le posizioni degli autisti, nonché i punti di ritiro e riconsegna in tempo quasi reale su una mappa per mostrare ai team di spedizione il quadro generale.
- Memorizza le ubicazioni degli agenti di consegna in modo da poter intervenire su di esse nella tua applicazione di backend o analizzarle nel tempo.
- Analizza la cronologia delle sedi per identificare tendenze e opportunità di ottimizzazione.



Di seguito è riportata una panoramica dei passaggi necessari per creare un'applicazione di distribuzione:

1. Crea le tue raccolte di geofence e collega i dispositivi tracciati alla raccolta. Per ulteriori informazioni, consulta [the section called “Geofencing e tracciamento”](#).
2. Crea una AWS Lambda funzione per aggiungere e rimuovere automaticamente i geofence man mano che i tuoi ordini vengono prenotati.
3. Configura Amazon EventBridge per inviare notifiche o avviare un processo. Per ulteriori informazioni, consulta [the section called “Reagire agli eventi con EventBridge”](#).
4. Visualizza gli asset tracciati e i geofence attivi su una mappa. [Per ulteriori informazioni, consulta Utilizzo delle mappe.](#)
5. Salva i dati sulla posizione in una memoria a lungo termine per ulteriori analisi.
6. Dopo aver creato la tua applicazione, puoi utilizzare Amazon CloudWatch e AWS CloudTrail gestirla. Per ulteriori informazioni, consulta [the section called “Monitoraggio con CloudWatch”](#) e [the section called “Utilizzo CloudTrail con Amazon Location”](#).

## Cos'è un fornitore di dati?

Usa Amazon Location Service per accedere alle risorse di geolocalizzazione di più fornitori di dati tramite il tuo AWS account senza richiedere contratti o integrazioni di terze parti. Questo può aiutarti a concentrarti sulla creazione della tua applicazione, senza dover gestire account, credenziali, licenze e fatturazione di terze parti.

I seguenti servizi di localizzazione di Amazon utilizzano fornitori di dati.

- Mappe: scegli gli stili tra diversi fornitori di mappe quando [crei una risorsa cartografica](#). Puoi utilizzare le risorse cartografiche per creare una mappa interattiva per visualizzare i dati.
- Luoghi: scegli un fornitore di dati quando [crei una risorsa per l'indice dei luoghi](#) per supportare le query di geocodifica, geocodifica inversa e ricerche.
- Percorsi: [scegli un fornitore di dati per supportare le query per il calcolo degli itinerari in diverse aree geografiche e applicazioni quando crei una risorsa per il calcolo del percorso](#). Con il fornitore di dati scelto, Amazon Location Service ti consente di calcolare i percorsi in base ai dati della rete up-to-date stradale, ai dati sul traffico in tempo reale, alle chiusure pianificate e agli schemi storici del traffico.

Ogni provider raccoglie e cura i propri dati utilizzando mezzi diversi. Possono anche avere competenze diverse in diverse regioni del mondo. Questa sezione fornisce dettagli sui nostri fornitori di dati. Puoi selezionare qualsiasi fornitore di dati in base alle tue preferenze.

Assicurati di leggere i termini e le condizioni quando utilizzi i fornitori di dati di Amazon Location Service. Per ulteriori informazioni, consulta i [Termini di servizio AWS](#). Consulta anche la [the section called "Privacy dei dati"](#) sezione per ulteriori informazioni su come Amazon Location protegge la tua privacy.

## Copertura e funzionalità dei fornitori di dati

La tabella seguente mostra la copertura e le funzionalità di alto livello per ogni fornitore di dati.

Fornitore di dati	Copertura geografica	Copertura delle funzionalità	Regione AWS
Esri	Globale	Mappe, luoghi, percorsi	<a href="#">Tutte le regioni</a> in cui è disponibile Amazon Location.
Afferra	<a href="#">Sud-est asiatico</a>	Mappe, luoghi, percorsi	Solo Asia Pacifico (Singapore). ap-southeast-1
QUI	Globale	Mappe, luoghi, percorsi	<a href="#">Tutte le regioni</a> in cui è disponibile Amazon Location.
Open Data (Apri dati)	Globale	Mappe	<a href="#">Tutte le regioni</a> in cui è disponibile Amazon Location.

Per ulteriori informazioni sulle funzionalità specifiche di ciascun fornitore di dati, consulta [Funzionalità per fornitore di dati](#).

Ogni fornitore di dati raccoglie e produce dati in modi diversi. Puoi saperne di più sulle loro aree di copertura nei seguenti argomenti:

- [Copertura: Esri](#)
- [Copertura: Grab](#)
- [Copertura: QUI](#)
- [Copertura: Open Data](#)

Se riscontri un problema con i dati e desideri segnalare un errore al fornitore di dati, consulta i seguenti argomenti:

- [Segnalazione degli errori a Esri](#)
- [Segnalazione degli errori relativi ai dati GrabMaps](#)
- [Segnalazione degli errori a HERE](#)

- [Segnalazione di errori e contributo agli Open Data](#)

## Stili di mappa

Ogni fornitore di dati fornisce un set di stili di mappa per il rendering dei dati cartografici che fornisce. Ad esempio, uno stile potrebbe includere immagini satellitari o potrebbe essere ottimizzato per mostrare le strade per la navigazione. È possibile trovare l'elenco e gli esempi degli stili per ciascun provider nei seguenti argomenti.

- [Stili di mappe Esri](#)
- [Prendi gli stili delle mappe](#)
- [Stili delle mappe HERE](#)
- [Stili di mappa Open Data](#)

## Ulteriori informazioni su ciascun fornitore di dati

I seguenti collegamenti forniscono ulteriori informazioni su ciascun fornitore di dati.

- [Esri](#)
- [GrabMaps](#)
- [Tecnologie HERE](#)
- [Open Data \(Apri dati\)](#)

## Esri

Amazon Location Service utilizza i servizi di localizzazione di Esri per aiutare AWS i clienti a utilizzare mappe, geocodificare e calcolare i percorsi in modo efficace. I servizi di localizzazione di Esri sono realizzati con dati di ready-to-use localizzazione autorevoli e di alta qualità, curati da team esperti di cartografi, geografi e demografi.

Per ulteriori informazioni sulle funzionalità, vedere [Esri](#) sui provider di dati di Amazon Location Service.

### Argomenti

- [Stili di mappe Esri](#)
- [Copertura: Esri](#)

- [Termini di utilizzo e attribuzione dei dati: Esri](#)
- [Segnalazione degli errori a Esri](#)

## Stili di mappe Esri

Amazon Location Service supporta i seguenti stili di mappa Esri durante la [creazione di una risorsa cartografica](#).

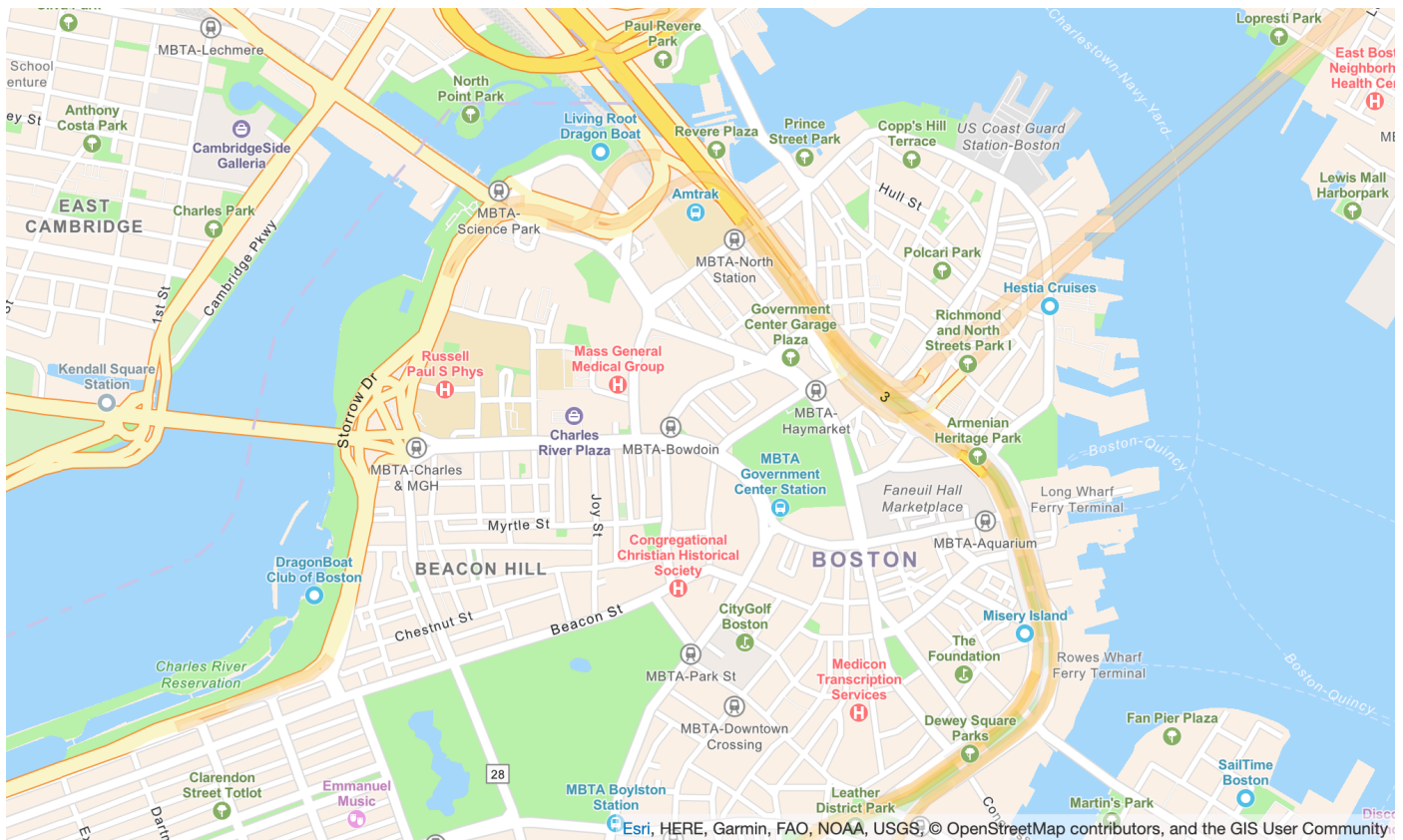
### Note

Gli stili di mappa Esri che non sono elencati in questa sezione non sono supportati.

Gli stili vettoriali Esri supportano alternative. [Opinioni politiche](#)

## Esri Navigation

### Navigazione Esri



Nome dello stile della mappa: VectorEsriNavigation

Questa mappa fornisce una mappa di base dettagliata del mondo, simboleggiata da uno stile di mappa di navigazione personalizzato progettato per essere utilizzato durante il giorno nei dispositivi mobili.

Questa mappa stradale completa include autostrade, strade principali, strade secondarie, ferrovie, giochi d'acqua, città, parchi, monumenti, impronte degli edifici e confini amministrativi. Il vector tile layer di questa mappa viene creato utilizzando le stesse fonti di dati utilizzate per World Street Map e altre mappe di base Esri. Abilita il POI layer impostandolo per sfruttare i dati aggiuntivi [CustomLayers](#) relativi ai luoghi.

Per ulteriori informazioni, vedere [Esri World Navigation sul sito Web](#) di Esri.

#### Note

La `VectorEsriNavigation` mappa illustrata sopra ha il POI layer abilitato.

## Fonts

Amazon Location fornisce caratteri utilizzando [GetMapGlyphs](#). Le seguenti sono le pile di caratteri disponibili per questa mappa:

- Arial Italic
- Arial Regolare
- Arial Bold
- Arial Unicode MS in grassetto
- Arial Unicode MS Normale

## Esri Imagery

### Immagini Esri

Nome dello stile della mappa: `RasterEsriImagery`

Questa mappa fornisce immagini satellitari e aeree di almeno un metro in molte parti del mondo e immagini satellitari a bassa risoluzione in tutto il mondo.

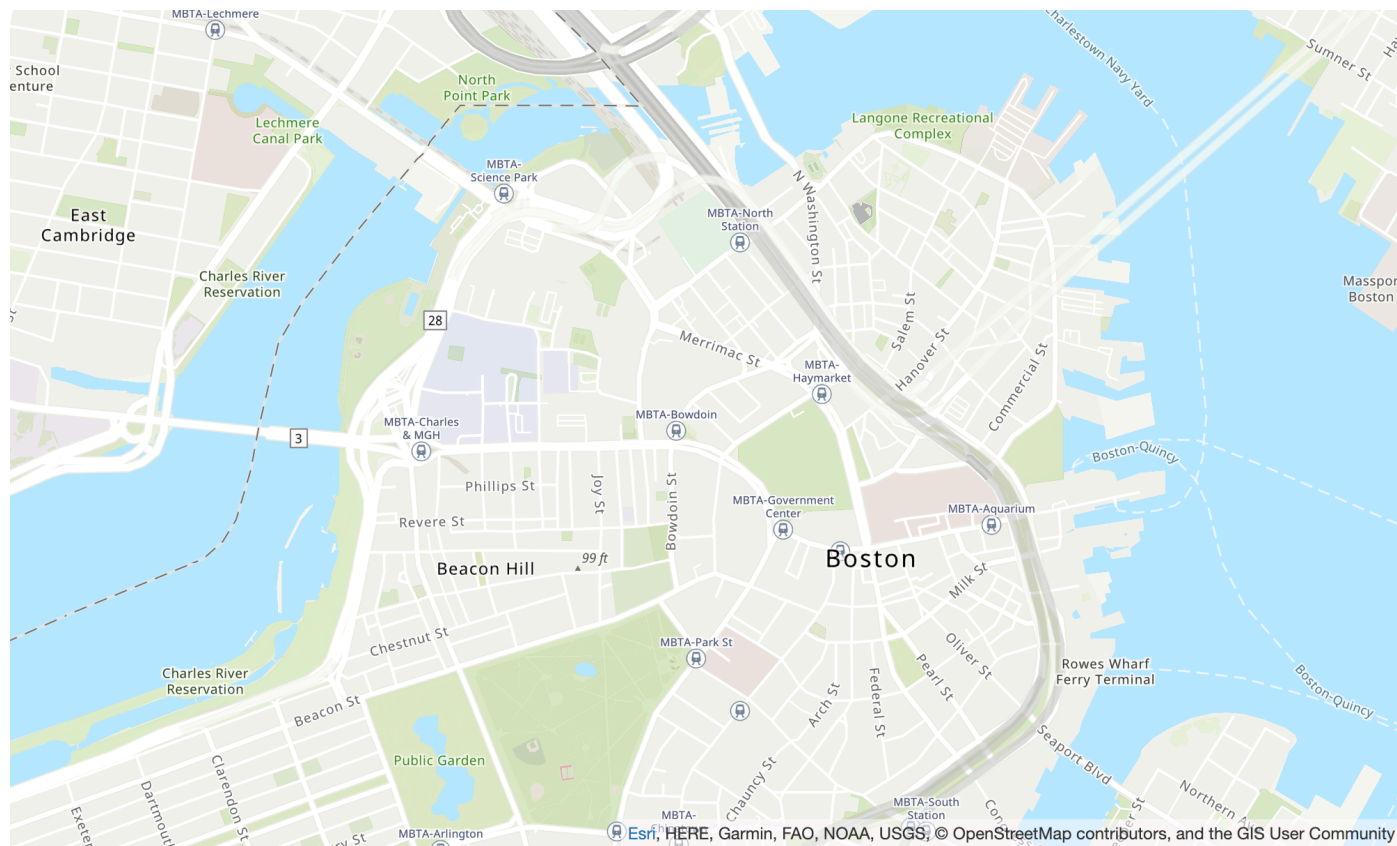
La mappa include immagini di 15 milioni di immagini su piccola e media scala (~ 1:591 M fino a ~ 1:72 k) e immagini SPOT di 2,5 m (da ~ 1:288 k a ~ 1:72 k) per il mondo. La mappa presenta

immagini con una risoluzione di 0,5 m negli Stati Uniti continentali e in alcune parti dell'Europa occidentale scattate da Maxar. Questa mappa presenta immagini submetriche aggiuntive di Maxar in molte parti del mondo. In altre parti del mondo, la GIS User Community ha contribuito con immagini a diverse risoluzioni. In alcune comunità, sono disponibili immagini ad altissima risoluzione (fino a 0,03 m) in scala ~ 1:280.

Per ulteriori informazioni, vedere [Esri World Imagery sul sito Web di Esri](#).

## Esri Light

### Esri Light



Nome dello stile della mappa: VectorEsriTopographic

Ciò fornisce una mappa di base dettagliata per il mondo simboleggiata con uno stile di mappa Esri classico. Ciò include autostrade, strade principali, strade secondarie, ferrovie, giochi d'acqua, città, parchi, monumenti, impronte degli edifici e confini amministrativi.

Questa mappa di base è compilata da una serie di fonti autorevoli provenienti da diversi fornitori di dati, tra cui US Geological Survey (USGS), US Environmental Protection Agency (EPA), US National Park Service (NPS), Organizzazione per l'alimentazione e l'agricoltura delle Nazioni Unite (FAO), Department of Natural Resources Canada (NRCAN), HERE ed Esri. I dati per aree



selezionate provengono dai contributori. OpenStreetMap Inoltre, i dati sono forniti dalla comunità GIS.

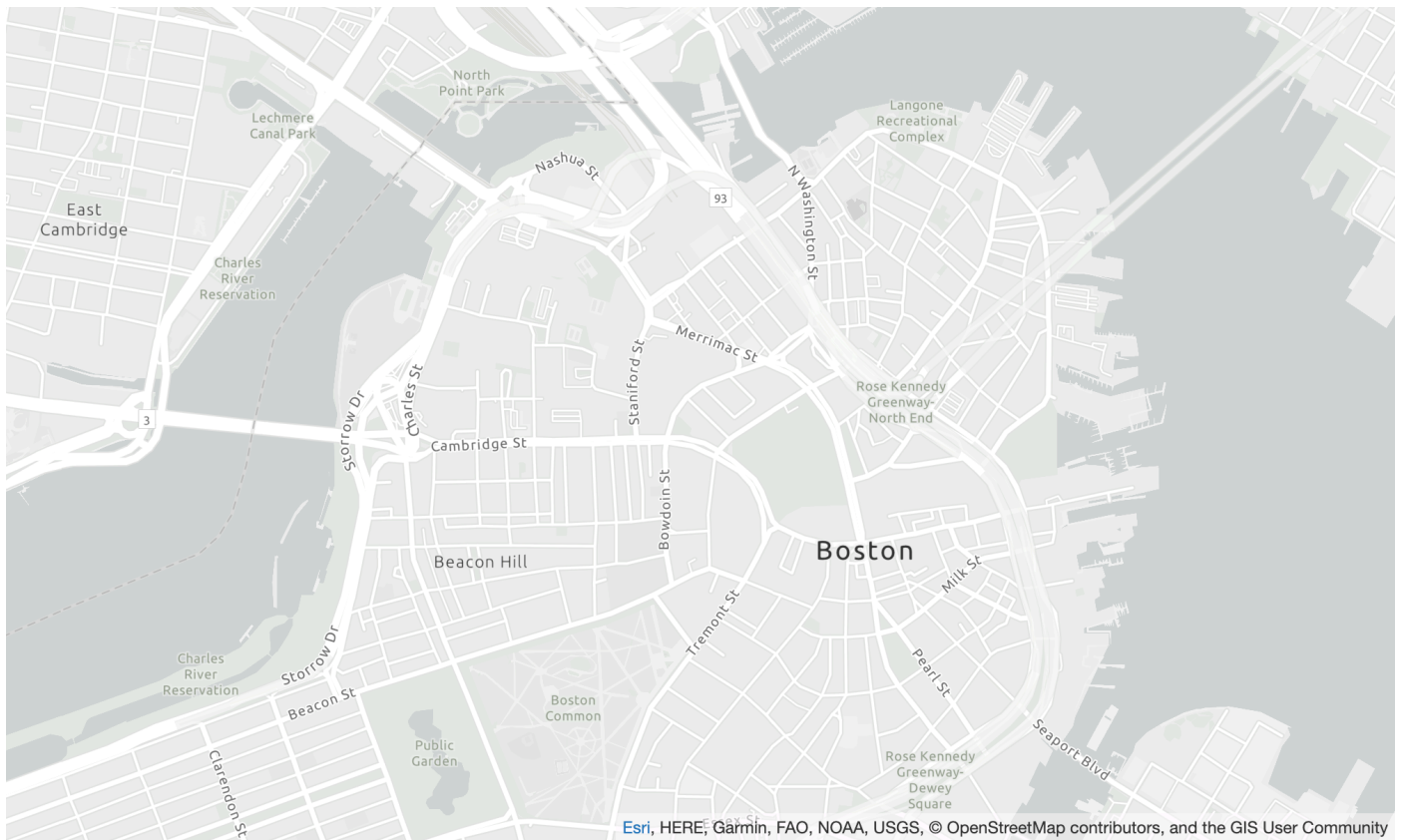
## Fonts

Amazon Location fornisce caratteri utilizzando [GetMapGlyphs](#). Le seguenti sono le pile di caratteri disponibili per questa mappa:

- Noto Sans Italic
- Noto Sans regolare
- Noto Sans Bold
- Noto Serif regolare
- Luce condensata Roboto Italic

## Esri Light Gray Canvas

### Tela Esri grigio chiaro



Nome dello stile della mappa: `VectorEsriLightGrayCanvas`

Questa mappa fornisce una mappa di base dettagliata del mondo simboleggiata da uno stile di sfondo neutro grigio chiaro con colori, etichette e caratteristiche minimali, progettata per attirare l'attenzione sui contenuti tematici.

Questo layer di tile vettoriali è creato utilizzando le stesse fonti di dati utilizzate per Light Grey Canvas e altre mappe di base Esri. La mappa include autostrade, strade principali, strade secondarie, ferrovie, giochi d'acqua, città, parchi, monumenti, impronte degli edifici e confini amministrativi.

Per ulteriori informazioni, vedere [Esri Light Grey Canvas](#) sul sito Web di Esri.

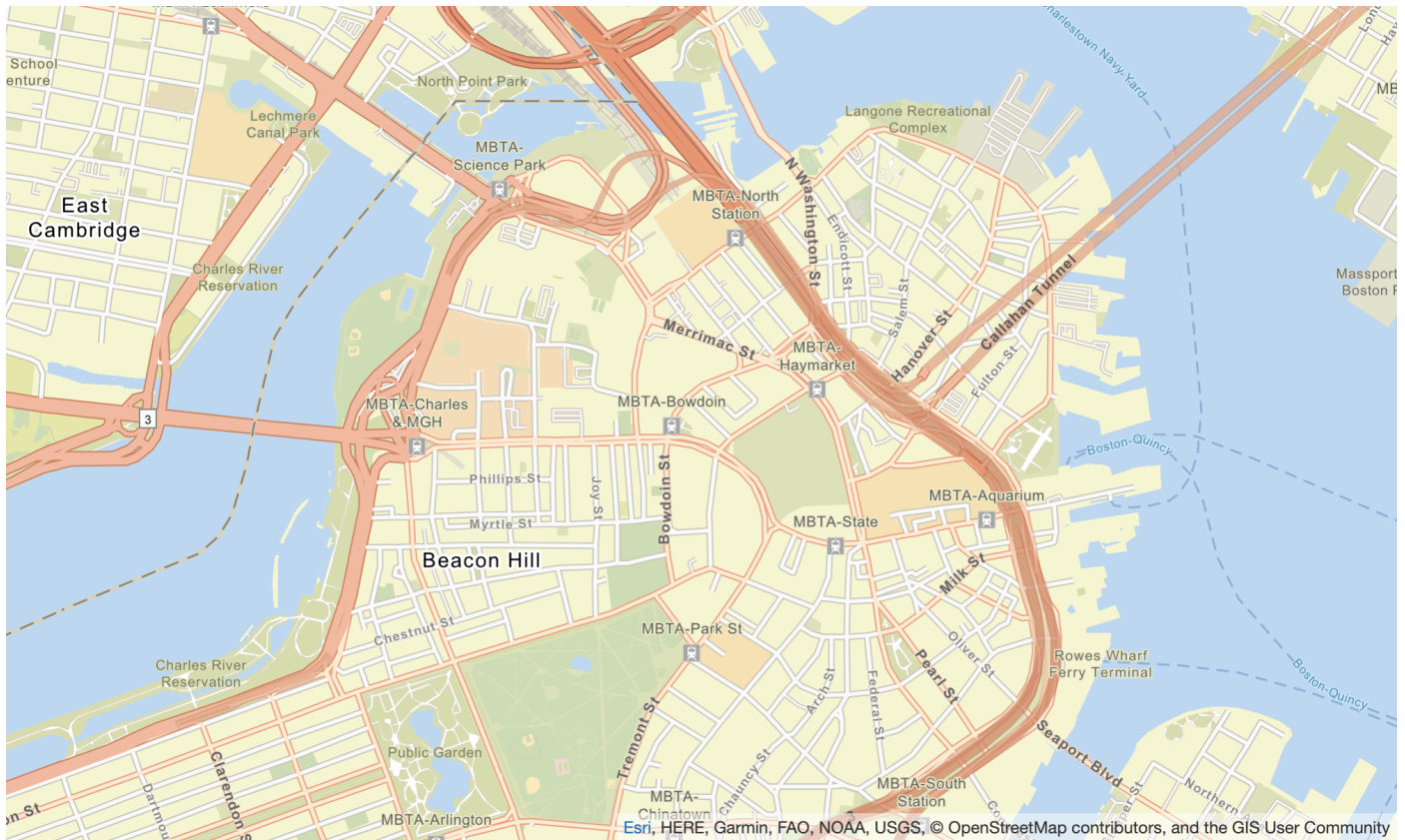
## Fonts

Amazon Location fornisce caratteri utilizzando [GetMapGlyphs](#). Le seguenti sono le pile di caratteri disponibili per questa mappa:

- Ubuntu Italic
- Ubuntu normale
- Ubuntu Light
- Ubuntu Bold

## Esri Street Map

Mappa stradale di Esri



## Nome dello stile della mappa: VectorEsriStreets

Questa mappa fornisce una mappa di base dettagliata del mondo, simboleggiata da uno stile di mappa di navigazione personalizzato progettato per essere utilizzato durante il giorno nei dispositivi mobili.

Questa mappa stradale completa include autostrade, strade principali, strade secondarie, ferrovie, giochi d'acqua, città, parchi, monumenti, impronte degli edifici e confini amministrativi. Include anche una serie più ricca di luoghi, come negozi, servizi, ristoranti, attrazioni e altri punti di interesse. Il vector tile layer di questa mappa viene creato utilizzando le stesse fonti di dati utilizzate per World Street Map e altre mappe di base Esri.

Per ulteriori informazioni, vedere [Esri World Street sul sito Web di Esri](#).

## Fonts

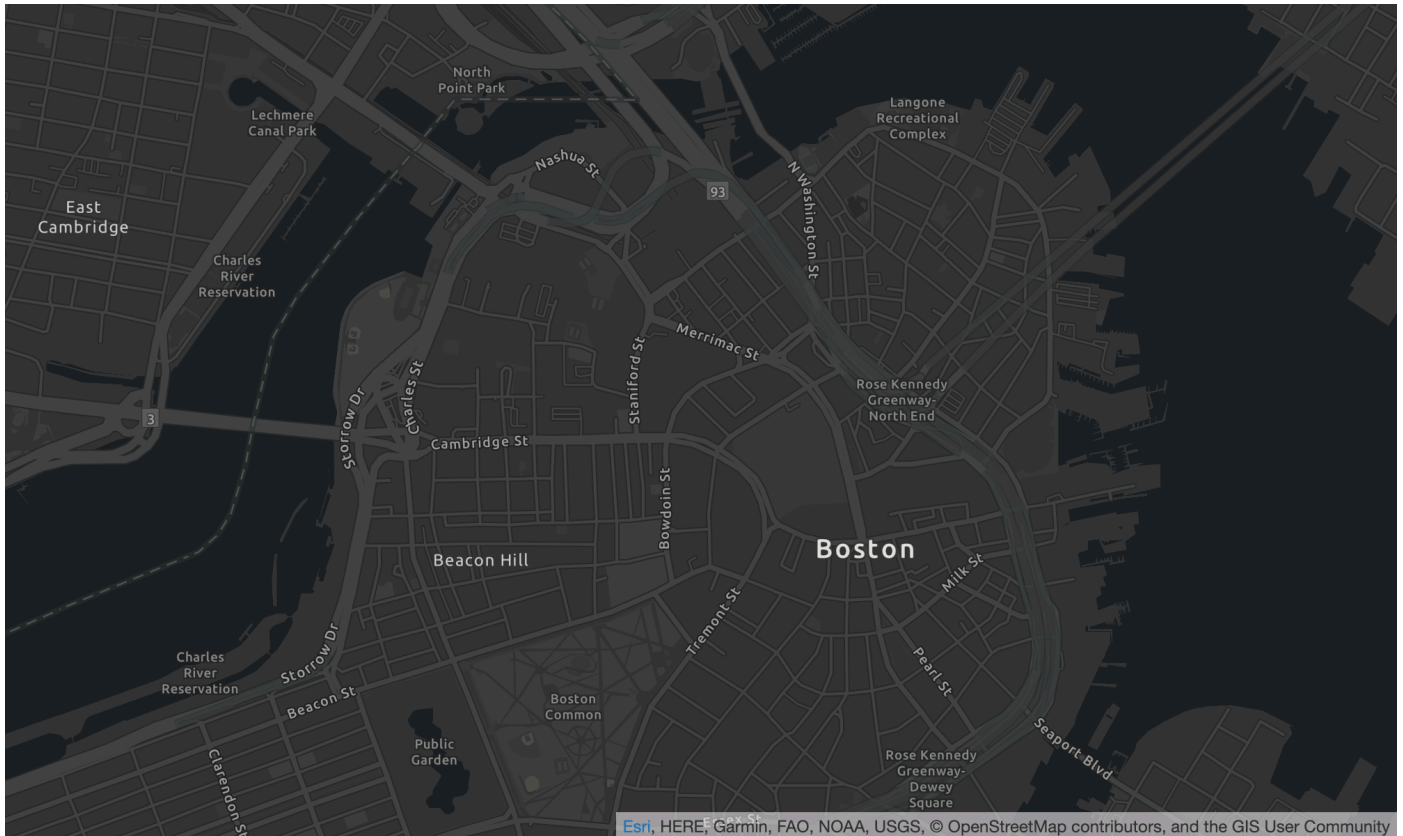
Amazon Location fornisce caratteri utilizzando [GetMapGlyphs](#). Le seguenti sono le pile di caratteri disponibili per questa mappa:

- Arial Italic

- Arial Regolare
- Arial Bold
- Arial Unicode MS in grassetto
- Arial Unicode MS Normale

## Esri Dark Gray Canvas

### Tela Esri grigio scuro



Nome dello stile della mappa: `VectorEsriDarkGrayCanvas`

Questa mappa fornisce una mappa di base vettoriale dettagliata del mondo, simboleggiata da uno stile di sfondo neutro grigio scuro con colori, etichette e funzionalità minimali, progettata per attirare l'attenzione sui contenuti tematici.

Questa mappa include autostrade, strade principali, strade secondarie, ferrovie, giochi d'acqua, città, parchi, monumenti, impronte degli edifici e confini amministrativi. I layer vettoriali di questa mappa sono creati utilizzando le stesse fonti di dati utilizzate per la mappa raster Dark Grey Canvas e altre mappe di base Esri.

Per ulteriori informazioni, vedere [Esri Dark Gray Canvas sul sito Web di Esri](#).

## Fonts

Amazon Location fornisce caratteri utilizzando [GetMapGlyphs](#). Le seguenti sono le pile di caratteri disponibili per questa mappa:

- Ubuntu Medium Italic
- Ubuntu Medium
- Ubuntu Italic
- Ubuntu normale
- Ubuntu Bold

## Copertura: Esri

[È possibile utilizzare Esri come fornitore di dati per supportare query di geocodifica, geocodifica inversa e ricerche quando si crea una risorsa relativa all'indice dei luoghi o per supportare le query per il calcolo di un percorso quando si crea una risorsa per il calcolo del percorso.](#)

Esri offre diversi livelli di qualità dei dati in diverse regioni del mondo. Per ulteriori informazioni sulla copertura nella propria area di interesse, vedere:

- [Dettagli Esri sulla copertura di geocodifica](#)
- [Dettagli Esri sulle reti stradali e sulla copertura del traffico](#)

## Termini di utilizzo e attribuzione dei dati: Esri

Prima di utilizzare i dati di Esri, assicuratevi di poter rispettare tutti i requisiti legali applicabili, incluse le condizioni di licenza applicabili a Esri e AWS.

Per ulteriori informazioni sui AWS requisiti, consulta i [Termini di servizio AWS](#).

Per informazioni sulle linee guida di attribuzione di Esri, consulta le [attribuzioni dei dati e le condizioni d'uso](#) di Esri.

## Segnalazione degli errori a Esri

Se si riscontra un problema con i dati e si desidera segnalare errori e discrepanze a Esri, consultare l'articolo del supporto tecnico di Esri relativo [a Come: fornire feedback sulle mappe di base e sulla geocodifica](#).

## GrabMaps

Grab è la più grande organizzazione di consegne nel sud-est asiatico, con milioni di autisti partner e clienti. La loro filiale [GrabMaps](#), crea dati up-to-date cartografici in quei paesi/regioni per uso proprio e per altri. Amazon Location Service utilizza i GrabMaps «servizi di localizzazione» per aiutare AWS i clienti a utilizzare mappe, geocodificare e calcolare i percorsi in modo efficace. GrabMaps«I servizi di localizzazione sono progettati per fornire dati di ready-to-use localizzazione autorevoli e di alta qualità, in particolare per i paesi del sud-est asiatico.

Per informazioni sulle funzionalità aggiuntive, consulta i [GrabMaps](#)provider di dati di Amazon Location Service.

### Important

Grab fornisce mappe solo per il sud-est asiatico ed è disponibile solo nella regione Asia Pacifico (Singapore) (ap-southeast-1). Per ulteriori informazioni, consulta [Paesi/regioni e aree coperte](#).

### Argomenti

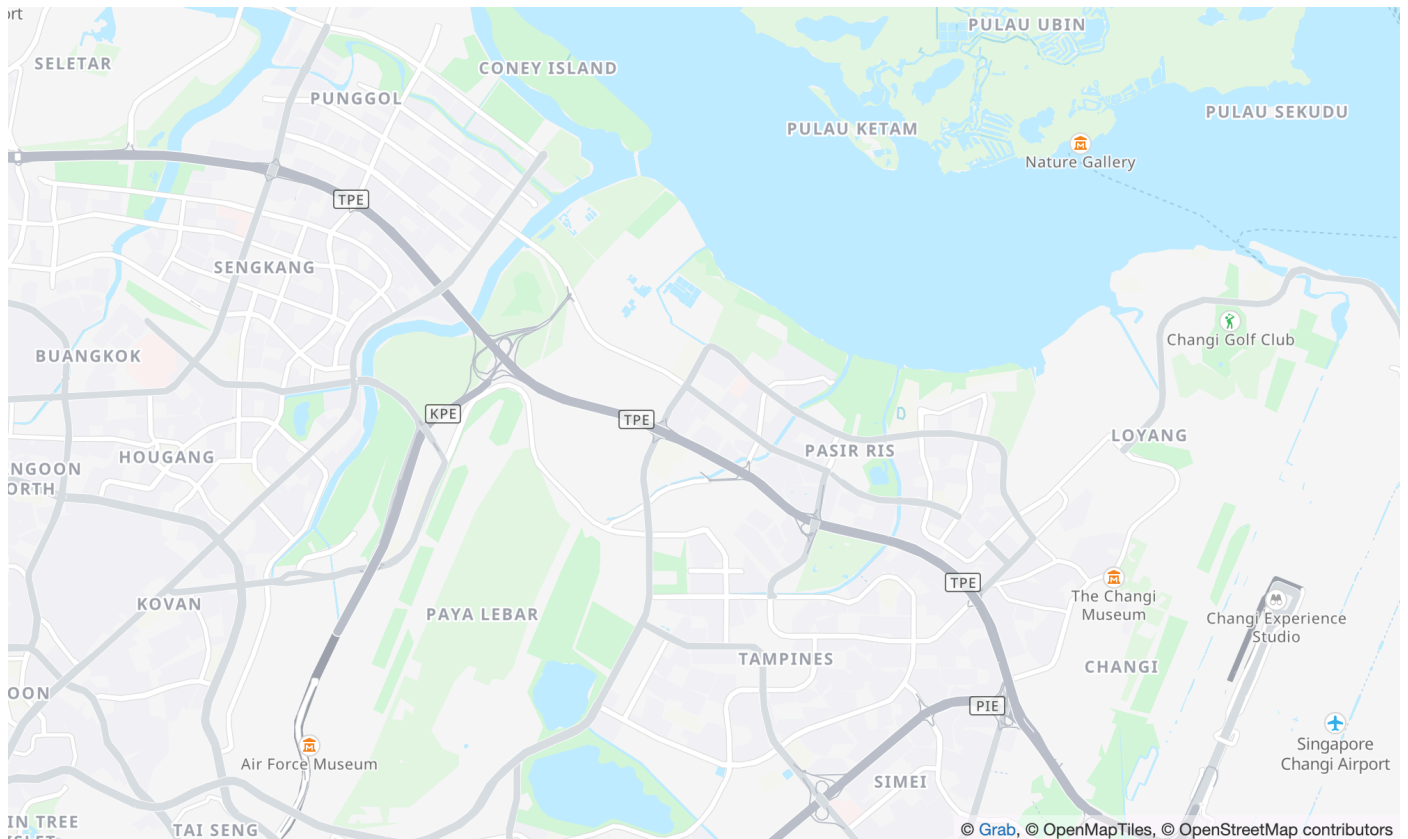
- [Prendi gli stili delle mappe](#)
- [Copertura: Grab](#)
- [Paesi/regioni e aree coperte](#)
- [Termini di utilizzo e attribuzione dei dati: Grab](#)
- [Segnalazione degli errori relativi ai dati GrabMaps](#)

## Prendi gli stili delle mappe

Amazon Location Service supporta i seguenti stili di mappe Grab durante la [creazione di una risorsa cartografica](#):

**Note**

Gli stili di mappa Grab che non sono elencati in questa sezione non sono attualmente supportati.

**Grab Standard Light Map****Grab Standard Light Map**

Nome dello stile della mappa: `VectorGrabStandardLight`

Mappa di base standard di Grab con colorazione dettagliata dell'uso del suolo, nomi delle aree, strade, punti di riferimento e punti di interesse che coprono il sud-est asiatico.

**Fonts**

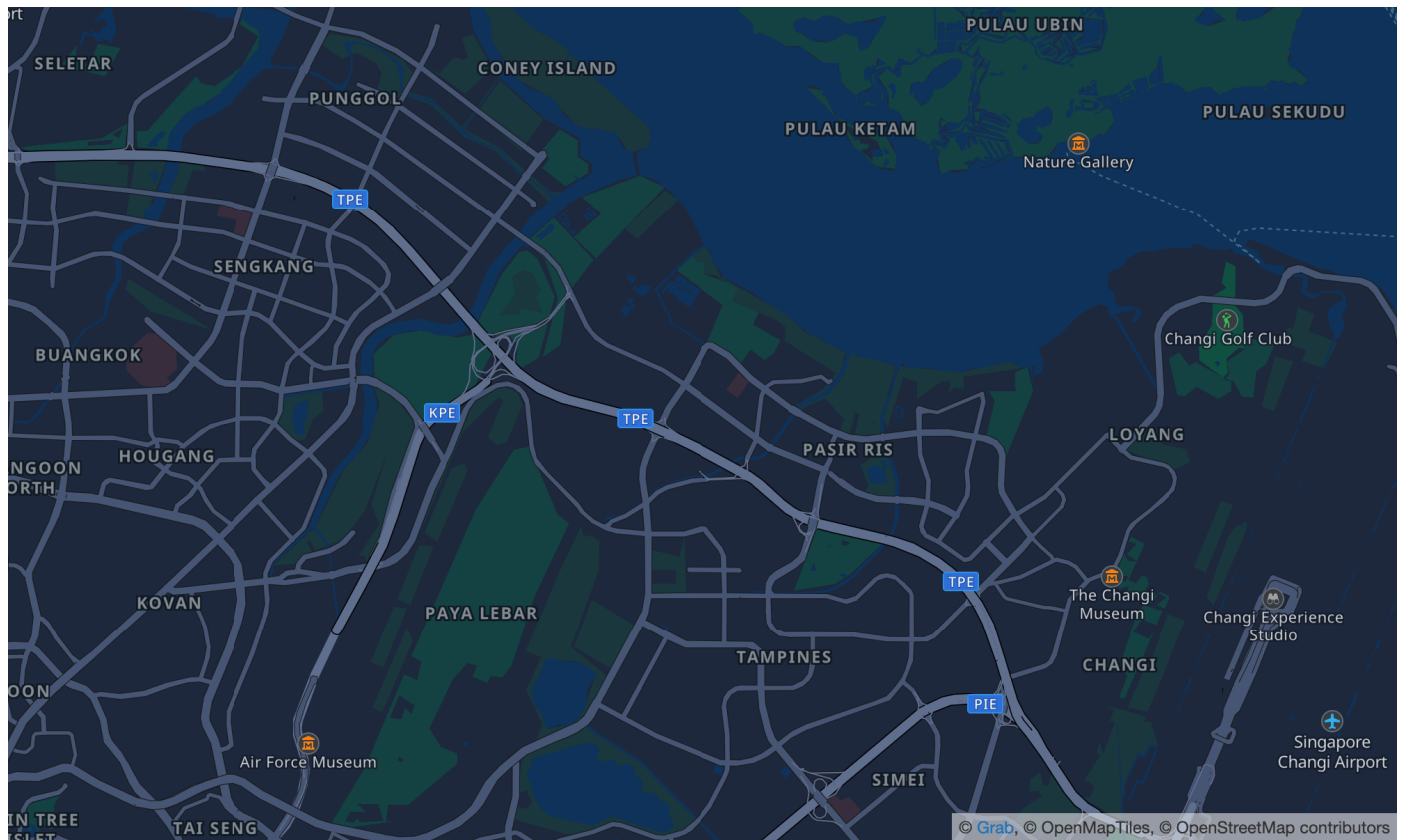
Amazon Location fornisce caratteri utilizzando [GetMapGlyphs](#). Le seguenti sono le pile di caratteri disponibili per questa mappa:

- Noto Sans Regular

- Noto Sans Medio
- Noto Sans Bold

## Grab Standard Dark Map

Prendi la mappa scura standard



Nome dello stile della mappa: VectorGrabStandardDark

La variante oscura di Grab della loro mappa di base standard, con colorazioni dettagliate sull'uso del suolo, nomi delle aree, strade, monumenti e punti di interesse che coprono il sud-est asiatico.

### Fonts

Amazon Location fornisce caratteri utilizzando [GetMapGlyphs](#). Le seguenti sono le pile di caratteri disponibili per questa mappa:

- Noto Sans Regular
- Noto Sans Medio
- Noto Sans Bold



## Copertura: Grab

È possibile utilizzare Grab come fornitore di dati per supportare le query per la geocodifica, la geocodifica inversa e le ricerche quando si crea una risorsa per l'indice dei luoghi o per supportare le query per il calcolo di un percorso quando si crea una risorsa per il calcolo del percorso.

### Paesi/regioni e aree coperte

Grab fornisce mappe solo per il sud-est asiatico ed è disponibile solo nella regione Asia Pacifico (Singapore) (). `ap-southeast-1`

Grab fornisce dati dettagliati per i seguenti paesi/regioni:

- Malesia
- Filippine
- Thailandia
- Singapore
- Vietnam
- Indonesia
- Birmania
- Cambogia

#### Note

Al di fuori di queste aree, le risorse di Amazon Location Service create con Grab come fornitore di dati non forniranno alcun risultato. Ciò include i risultati o i percorsi di ricerca.

Le mappe di Grab rientrano nei seguenti limiti:

- Sud — Latitudine -21.943045533438166
- Ovest — Longitudine 90,0
- Nord: latitudine 31.952162238024968
- Est: longitudine 146,25

Per i livelli di zoom da 1 a 4, Grab include una copertura globale. Per i livelli di zoom 5 e inferiori, i riquadri della mappa vengono forniti solo all'interno di questo riquadro limitato.

#### Note

Al di fuori di questo riquadro limitato, le risorse cartografiche di Amazon Location Service create con Grab come fornitore di dati non restituiranno riquadri di mappe. Per evitare di visualizzare 404 errori nella tua applicazione, puoi limitare la mappa con un riquadro di delimitazione, come descritto in [Impostazione delle estensioni per una mappa utilizzando MapLibre](#)

### Modalità di viaggio Grab routing

Per quanto riguarda il routing, Grab fornisce il routing per auto e moto per tutti i paesi/regioni elencati in precedenza.

Grab non supporta il routing dei camion.

Per percorsi ciclabili e pedonali, Grab supporta le seguenti città:.

- Singapore
- Giacarta
- Manila
- Valle di Klang
- Bangkok
- Città di Ho Chi Minh
- Hanoi

### Termini di utilizzo e attribuzione dei dati: Grab

Quando si utilizzano i dati di Grab, è necessario rispettare tutti i requisiti legali applicabili, comprese le condizioni di licenza applicabili a Grab e. AWS

Per ulteriori informazioni sui AWS requisiti, consulta i [Termini di servizio AWS](#).

Per informazioni sulle GrabMaps «linee guida per l'attribuzione», consulta la Sezione 9.23 delle [attribuzioni dei dati e delle condizioni d'uso](#) di Grab.

## Segnalazione degli errori relativi ai dati GrabMaps

Se riscontri un problema con i dati di GrabMaps e desideri segnalare errori o discrepanze, [contatta l'AWS assistenza tecnica](#).

## Tecnologie HERE

Amazon Location Service utilizza i servizi di localizzazione di HERE Technologies per aiutare AWS i clienti a utilizzare mappe, geocodificare e calcolare i percorsi in modo efficace. I dati sulla posizione di HERE offrono una piattaforma incentrata sulla posizione aperta, sicura e privata. Selezionando i dati sulla posizione di HERE, stai selezionando dati accurati, aggiornati e affidabili che vengono distribuiti nativamente sul cloud. AWS

Per ulteriori informazioni sulle funzionalità, consulta [HERE](#) sui fornitori di dati di Amazon Location Service.

### Argomenti

- [Stili delle mappe HERE](#)
- [Copertura: QUI](#)
- [Termini di utilizzo e attribuzione dei dati: QUI](#)
- [Segnalazione degli errori a HERE](#)

## Stili delle mappe HERE

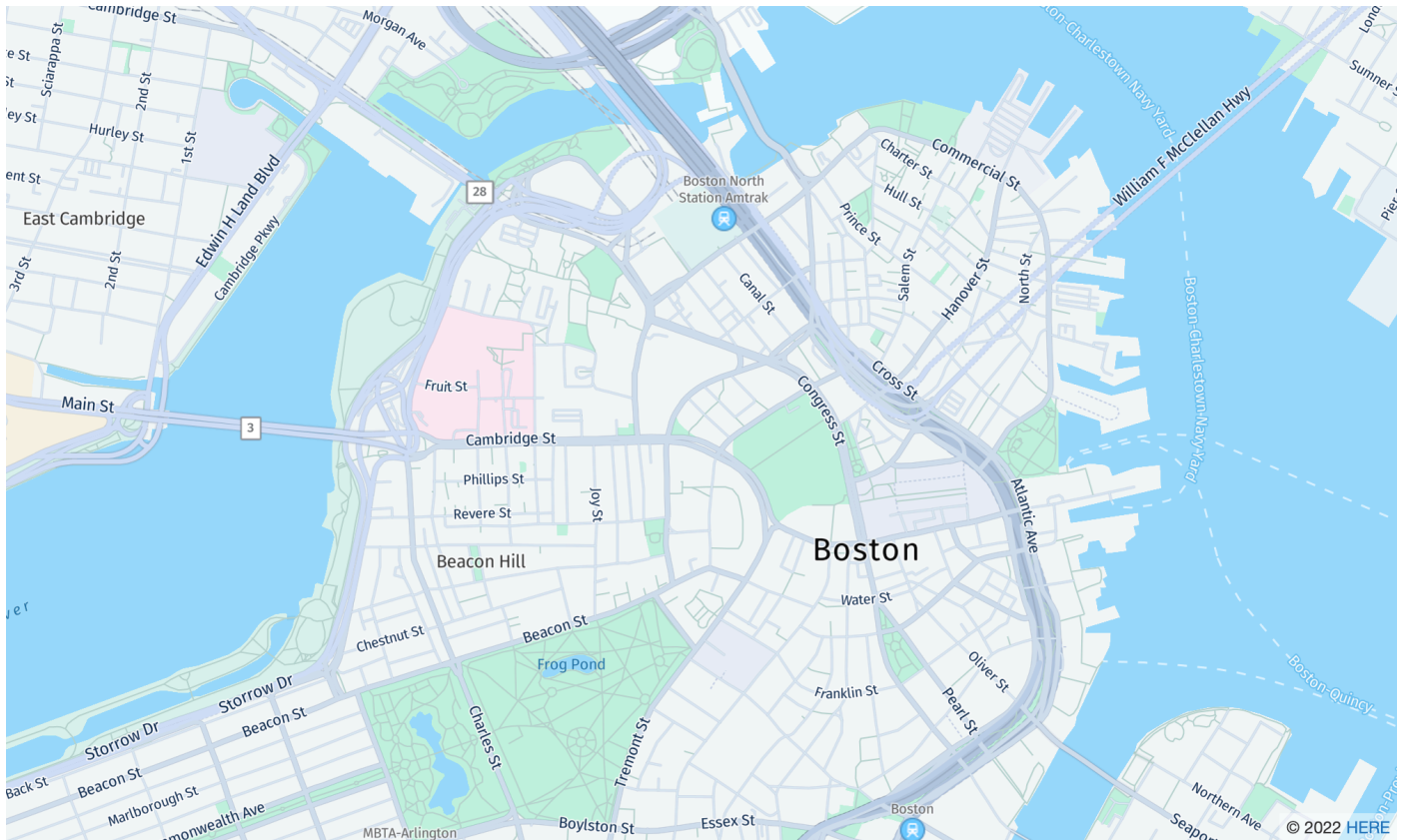
Amazon Location Service supporta i seguenti stili di mappa HERE durante la [creazione di una risorsa cartografica](#):

### Note

Gli stili di mappa HERE che non sono elencati in questa sezione non sono attualmente supportati.

## HERE Explore

HERE Esplora



Nome dello stile della mappa: VectorHereExplore

HERE Esplora

Una mappa base dettagliata e neutrale del mondo. La mappa stradale include autostrade, strade principali, strade secondarie, ferrovie, giochi d'acqua, città, parchi, monumenti, impronte degli edifici e confini amministrativi. Include una mappa del Giappone completamente progettata.

Fonts

Amazon Location fornisce caratteri utilizzando [GetMapGlyphs](#). Le seguenti sono le pile di caratteri disponibili per questa mappa:

- Fira GO Italic
- Fira GO Regolare
- Fira GO Bold
- Lampada a infrarossi Noto Sans CJK
- Noto Sans CJP regolare
- Noto Sans CJP Gold

## HERE Imagery

### Immagini HERE



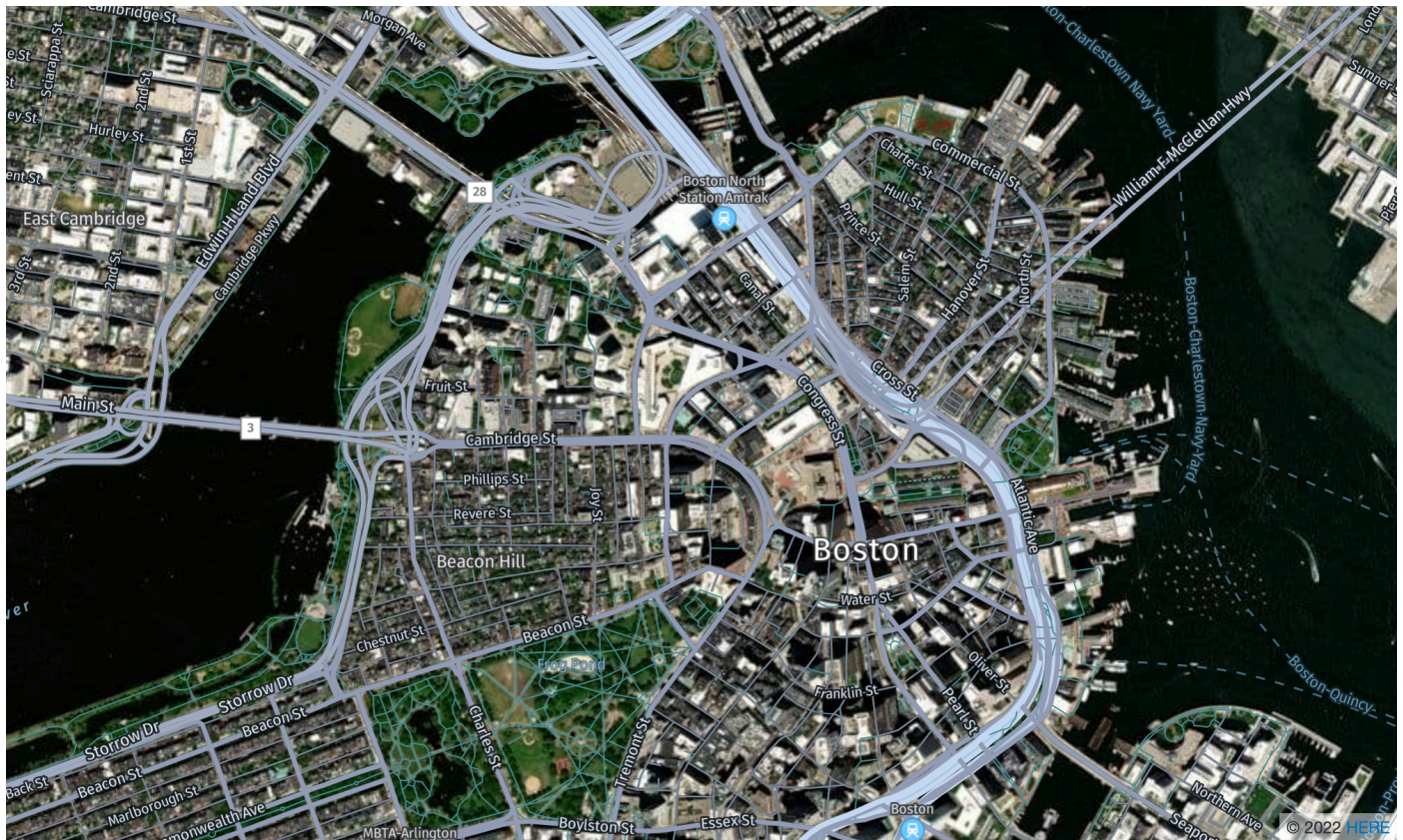
Nome dello stile della mappa: `RasterHereExploreSatellite`

### Immagini HERE

HERE Imagery fornisce immagini satellitari ad alta risoluzione con copertura globale.

### HERE Hybrid

### HERE Hybrid



Nome dello stile della mappa: HybridHereExploreSatellite

HERE Hybrid

HERE Hybrid style mostra la rete stradale, i nomi delle strade e le etichette delle città su immagini satellitari. Questo stile sovrappone due riquadri della mappa: l'immagine satellitare (riquadro raster) sullo sfondo e la rete stradale e le etichette (riquadro vettoriale) nella parte superiore. Questo stile recupererà automaticamente i riquadri raster e vettoriali necessari per il rendering della mappa.

#### Note

Gli stili ibridi utilizzano sia i riquadri vettoriali che quelli raster durante il rendering della mappa visualizzata. Ciò significa che vengono recuperati più riquadri rispetto all'utilizzo esclusivo di riquadri vettoriali o raster. I costi includeranno tutti i riquadri recuperati.

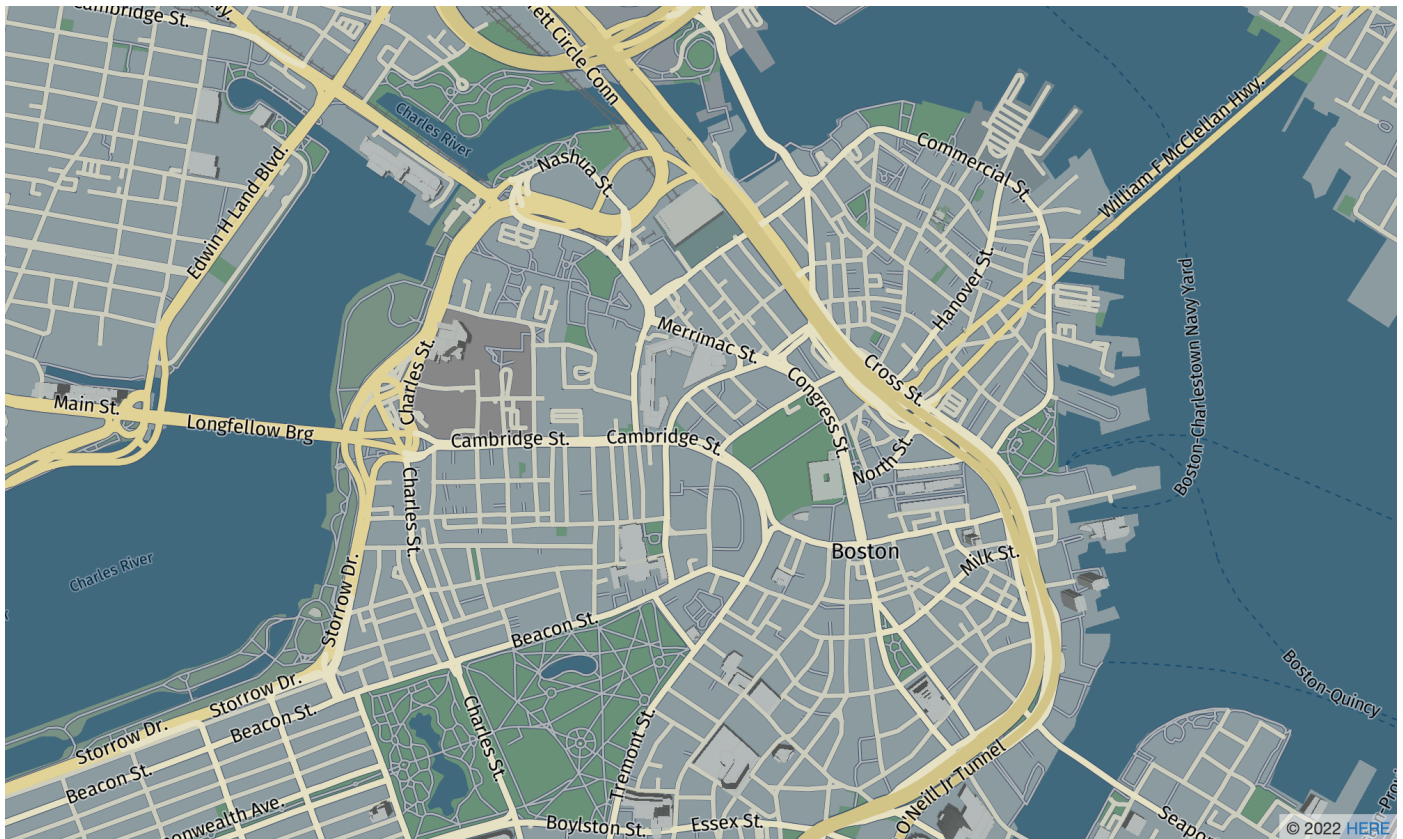
Fonts

Amazon Location fornisce caratteri utilizzando [GetMapGlyphs](#). Le seguenti sono le pile di caratteri disponibili per questa mappa:

- Fira GO Italic
- Fira GO Regolare
- Fira GO Bold
- Lampada a infrarossi Noto Sans CJK
- Noto Sans CJP regolare
- Noto Sans CJP Gold

HERE Contrast (Berlin)

HERE Contrast (Berlino)



Nome dello stile della mappa: VectorHereContrast

HERE Contrast (Berlino)

Una mappa base dettagliata del mondo che unisce il rendering 3D e 2D. La mappa stradale ad alto contrasto include autostrade, strade principali, strade secondarie, ferrovie, giochi d'acqua, città, parchi, monumenti, impronte degli edifici e confini amministrativi.

## Fonts

Amazon Location fornisce caratteri utilizzando [GetMapGlyphs](#). Le seguenti sono le pile di caratteri disponibili per questa mappa:

- Fira GO Regular
- Fira GO Bold

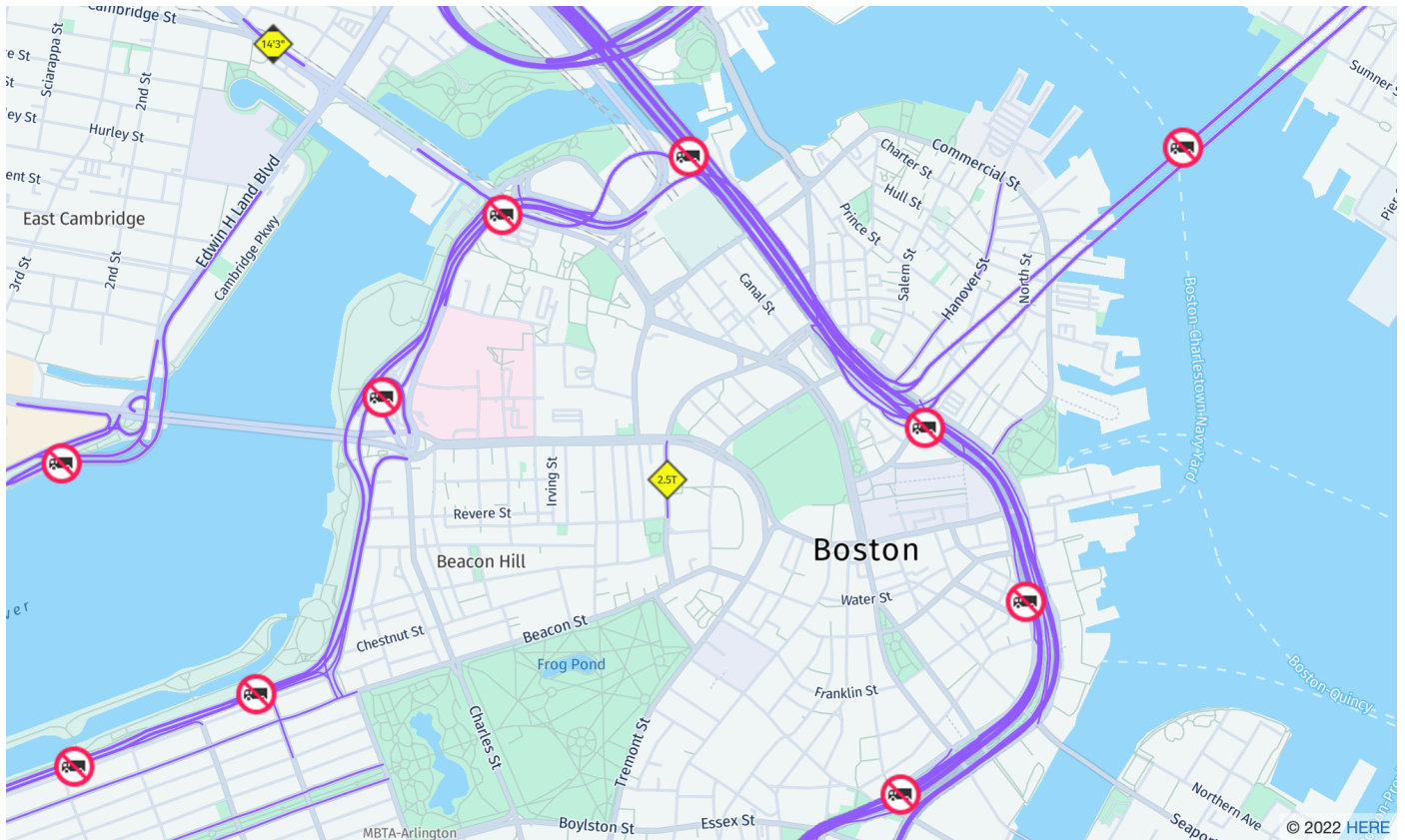
### Note

Questo stile è stato rinominato da `VectorHereBerlin` (HERE Berlin maps). `VectorHereBerlin` è obsoleto, ma continuerà a funzionare nelle applicazioni che lo utilizzano.

HERE Explore Truck

QUI Esplora Truck





Nome dello stile della mappa: VectorHereExploreTruck

HERE Explore Truck

Una mappa base dettagliata e neutrale del mondo. La mappa stradale si basa sullo stile HERE Explore ed evidenzia le restrizioni e gli attributi dei binari (tra cui larghezza, altezza e HAZMAT) con simboli e icone, per supportare i casi d'uso nel settore dei trasporti e della logistica.

Fonts

Amazon Location fornisce caratteri utilizzando [GetMapGlyphs](#). Le seguenti sono le pile di caratteri disponibili per questa mappa:

- Fira GO Italic
- Fira GO Regolare
- Fira GO Bold
- Lampada a infrarossi Noto Sans CJK
- Noto Sans CJP regolare
- Noto Sans CJP Gold

Per ulteriori informazioni sulla qualità dei dati cartografici in diverse regioni del mondo, consulta la copertura della mappa [HERE](#).

## Copertura: QUI

[Puoi utilizzare HERE come fornitore di dati per supportare le query per la geocodifica, la geocodifica inversa e le ricerche quando crei una risorsa per l'indice dei luoghi o per supportare le query per il calcolo di un percorso quando crei una risorsa per il calcolo del percorso.](#)

HERE offre diversi livelli di qualità dei dati in diverse regioni del mondo. Per ulteriori informazioni sulla copertura nella tua area di interesse, consulta quanto segue:

- [Copertura di geocodifica HERE](#)
- [Copertura del routing automobilistico HERE](#)
- [Copertura del routing dei camion HERE](#)

## Termini di utilizzo e attribuzione dei dati: QUI

Prima di utilizzare i dati HERE, assicurati di poter rispettare tutti i requisiti legali applicabili, incluse le condizioni di licenza applicabili a HERE e AWS. A causa delle limitazioni di licenza, non puoi utilizzare HERE per archiviare i risultati di geocodifica per località in Giappone.

Per informazioni sui AWS requisiti, consulta i [termini dei servizi AWS](#).

Per ulteriori informazioni sulle linee guida di attribuzione di HERE, consulta la Sezione 2 dei [Termini per i fornitori di HERE Technologies applicabili alla posizione e ad](#) altri contenuti.

## Segnalazione degli errori a HERE

Per segnalare errori e discrepanze nelle mappe a HERE, vai su <https://www.here.com/contact> e scegli Segnala un errore di mappa.

## Open Data (Apri dati)

Amazon Location Service fornisce l'accesso ai dati cartografici open source tramite il provider Open Data. Open Data fornisce mappe di base globali basate [sulla distribuzione di mappe Daylight](#) di [OpenStreetMap \(OSM\)](#), [Natural Earth](#) e altre fonti di dati aperte. Le mappe fornite sono progettate per supportare diverse applicazioni e casi d'uso, tra cui logistica e consegna e visualizzazione dei dati in ambienti web e mobili. Con oltre un milione di creatori di mappe, la comunità OSM aggiorna

centinaia di migliaia di funzionalità al giorno. Amazon Location Service incorpora regolarmente queste modifiche.

Per ulteriori informazioni sulle funzionalità, consulta [Open Data](#) sui provider di dati di Amazon Location Service.

## Argomenti

- [Stili di mappa Open Data](#)
- [Copertura: Open Data](#)
- [Termini di utilizzo e attribuzione dei dati: Open Data](#)
- [Segnalazione di errori e contributo agli Open Data](#)

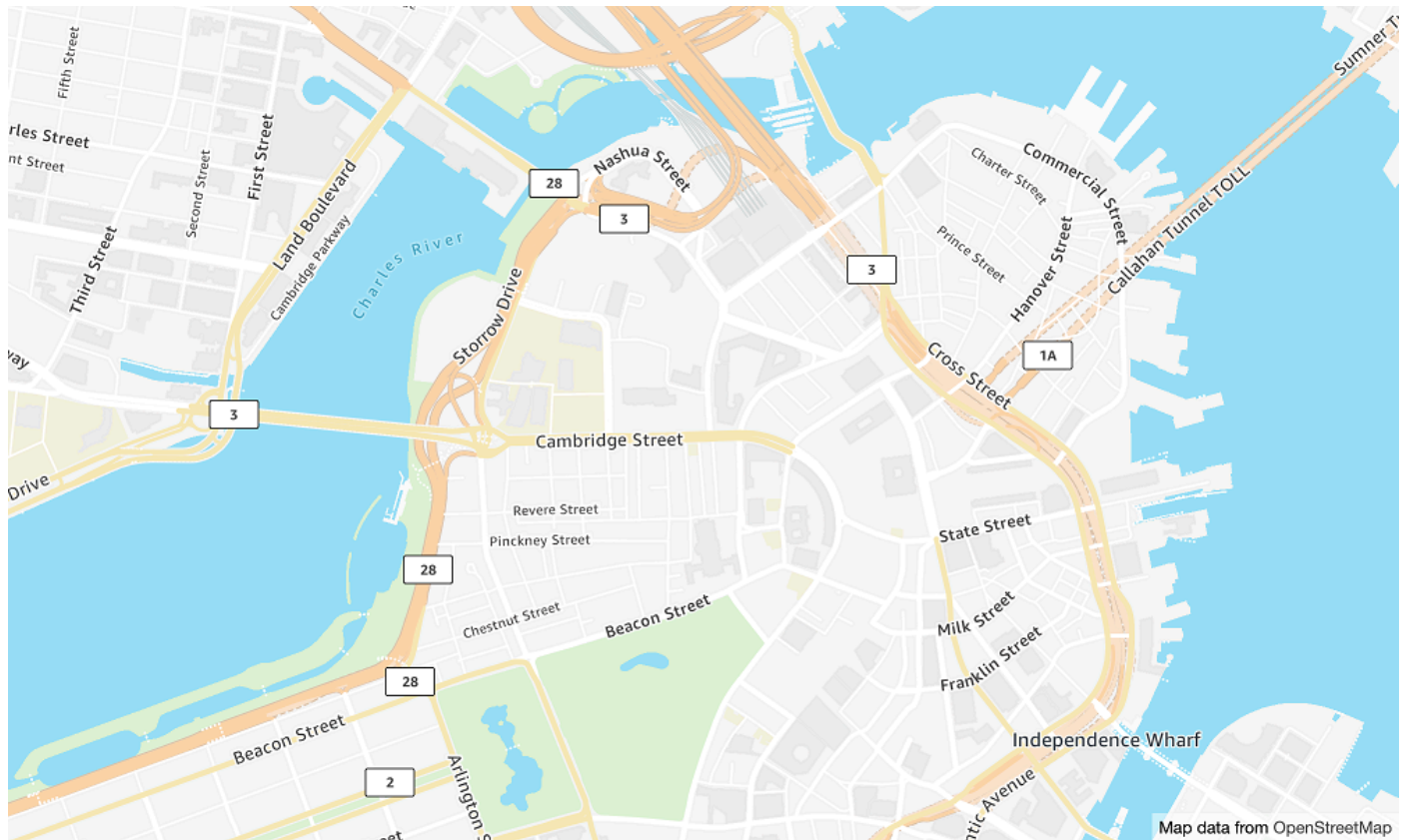
## Stili di mappa Open Data

Amazon Location Service supporta i seguenti stili di mappa durante la [creazione di una risorsa cartografica](#):

Gli stili di mappa Open Data supportano alternative [Opinioni politiche](#).

### Open Data Standard Light

Open Data Standard Light



Nome dello stile della mappa: VectorOpenDataStandardLight

Ciò fornisce una mappa di base dettagliata del mondo in uno stile di mappa leggero, adatto all'uso di siti Web e applicazioni mobili. Ciò include autostrade, strade principali, strade secondarie, ferrovie, giochi d'acqua, città, parchi, monumenti, impronte degli edifici e confini amministrativi.

Questa mappa di base si basa sulla [distribuzione delle mappe OSM Daylight compilata dai OpenStreetMap contributori](#) (OSM). La comunità OSM include oltre 1,8 milioni di collaboratori che aggiornano più di 500.000 funzionalità ogni giorno. Amazon Location Service incorpora queste modifiche regolarmente.

## Fonts

Amazon Location fornisce caratteri utilizzando [GetMapGlyphs](#). Le seguenti sono le pile di caratteri disponibili per questa mappa:

- Amazon Ember Bold, Noto sans Bold
- Amazon Ember RC condensato Bold, Noto Sans Bold
- Amazon Ember RC condensato normale, Noto Sans normale

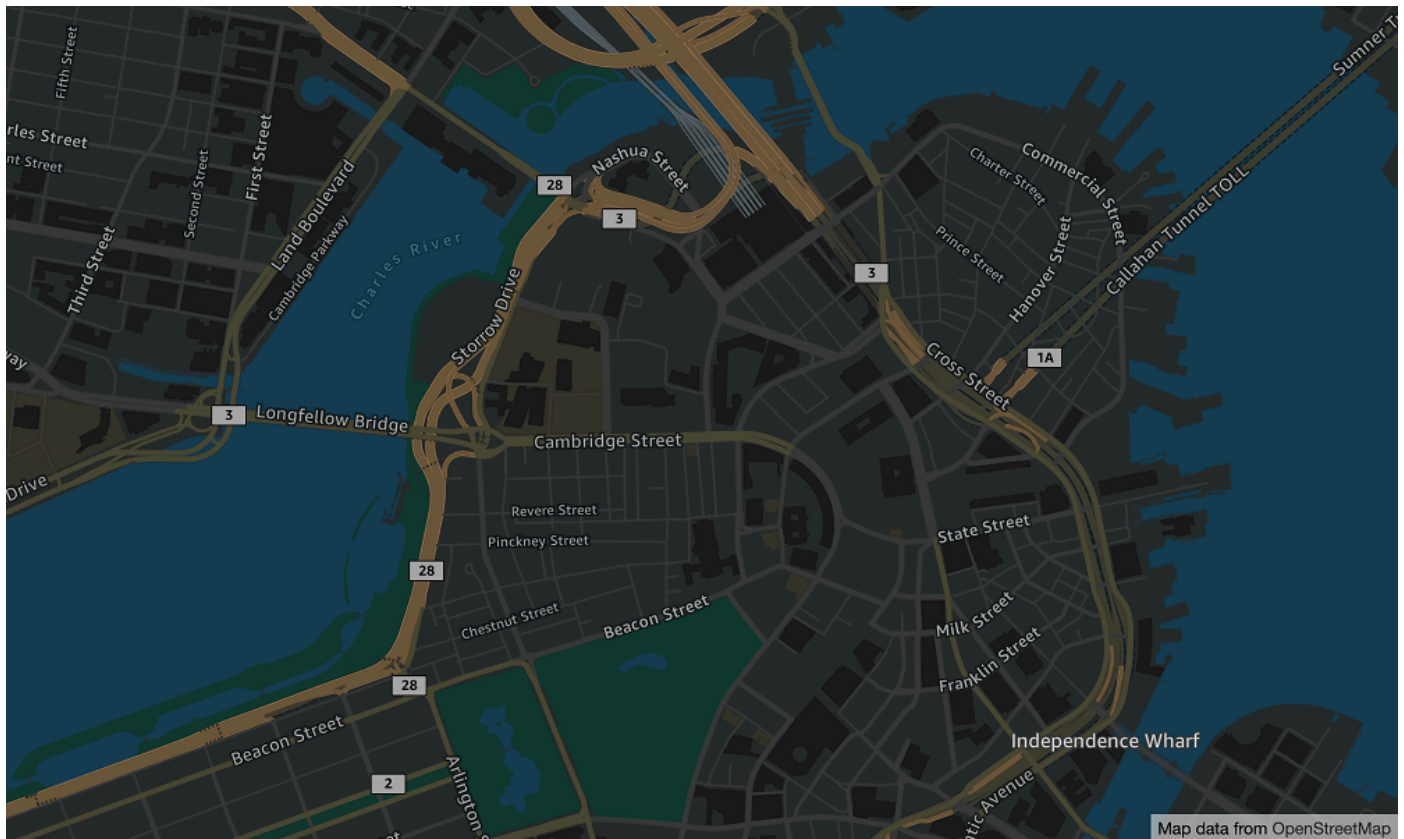
- Amazon Ember Medio, Noto Sans Medio
- Amazon Ember Regular Italic, Noto Sans Italic
- Amazon Ember Regolare, Noto Sans Regolare
- Amazon Ember Regolare, Noto Sans Regolare, Noto Sans Arabo Regolare
- Amazon Ember RC condensato Bold, Noto Sans Bold, Noto Sans Arabic Condensato Bold
- Amazon Ember Bold, Noto Sans Bold, Noto Sans Arabic Bold
- Amazon Ember Normale Italic, Noto Sans Italic, Noto Sans Arabo Regolare
- Amazon Ember RC condensato regolare, Noto Sans normale, Noto Sans Arabic Condensato Regolare
- Amazon Ember Medio, Noto Sans Medio, Noto Sans Arabic Medio

#### Note

I caratteri utilizzati da `VectorOpenDataStandardLight` sono caratteri combinati utilizzati per la maggior parte dei glifi ma `Amazon Ember Noto Sans` per i glifi non supportati da `Amazon Ember`

Open Data Standard Dark

Open Data Standard Dark



Nome dello stile della mappa: VectorOpenDataStandardDark

Si tratta di uno stile di mappa a tema scuro che fornisce una mappa di base dettagliata del mondo, adatta all'uso di siti Web e applicazioni mobili. Ciò include autostrade, strade principali, strade secondarie, ferrovie, giochi d'acqua, città, parchi, monumenti, impronte degli edifici e confini amministrativi.

Questa mappa di base si basa sulla [distribuzione delle mappe OSM Daylight compilata dai OpenStreetMap contributori](#) (OSM). La comunità OSM include oltre 1,8 milioni di collaboratori che aggiornano più di 500.000 funzionalità ogni giorno. Amazon Location Service incorpora queste modifiche regolarmente.

## Fonts

Amazon Location fornisce caratteri utilizzando [GetMapGlyphs](#). Le seguenti sono le pile di caratteri disponibili per questa mappa:

- Amazon Ember Bold, Noto sans Bold
- Amazon Ember RC condensato Bold, Noto Sans Bold
- Amazon Ember RC condensato normale, Noto Sans normale

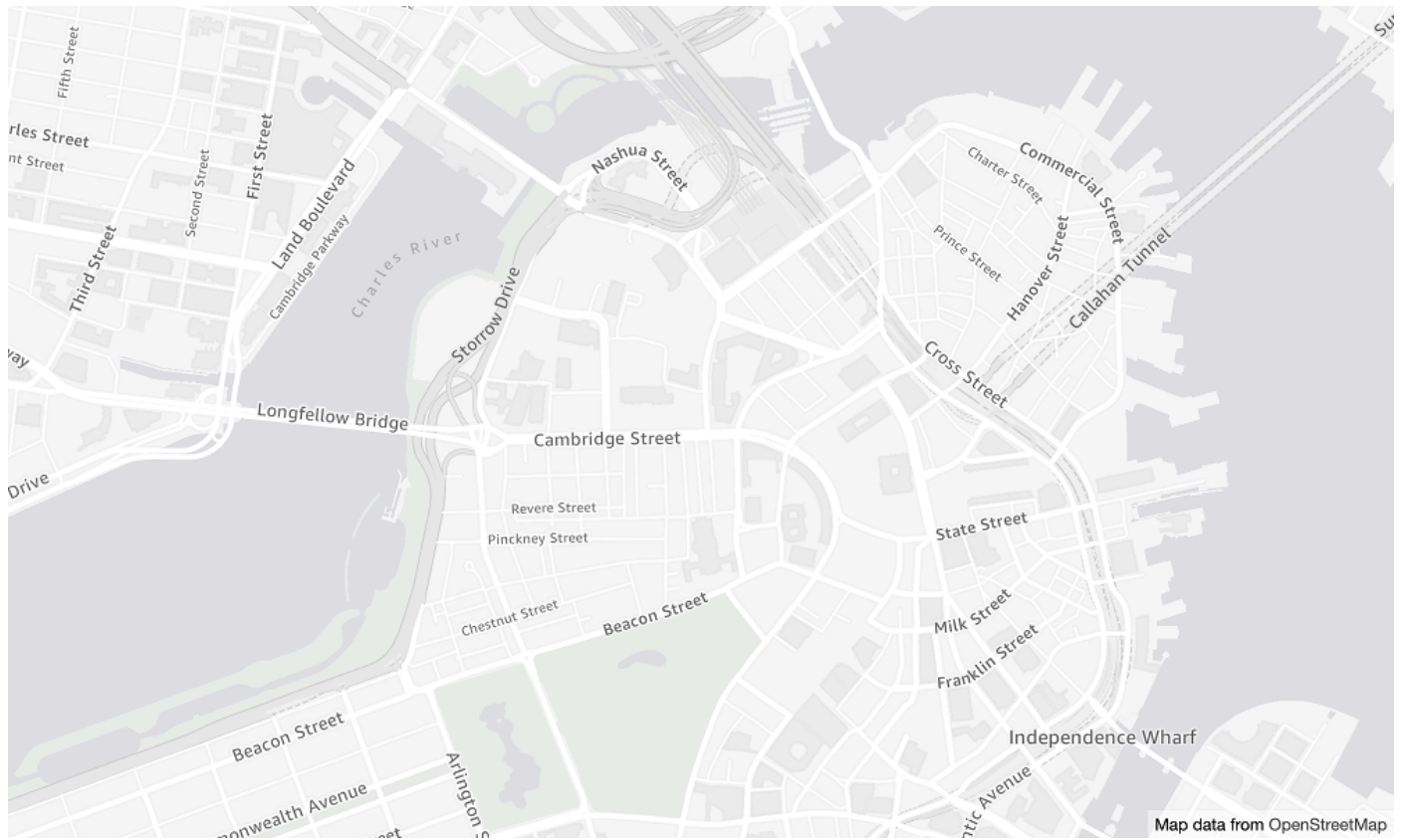
- Amazon Ember Medio, Noto Sans Medio
- Amazon Ember Regular Italic, Noto Sans Italic
- Amazon Ember Regolare, Noto Sans Regolare
- Amazon Ember Regolare, Noto Sans Regolare, Noto Sans Arabo Regolare
- Amazon Ember RC condensato Bold, Noto Sans Bold, Noto Sans Arabic Condensato Bold
- Amazon Ember Bold, Noto Sans Bold, Noto Sans Arabic Bold
- Amazon Ember Normale Italic, Noto Sans Italic, Noto Sans Arabo Regolare
- Amazon Ember RC condensato regolare, Noto Sans normale, Noto Sans Arabic Condensato Regolare
- Amazon Ember Medio, Noto Sans Medio, Noto Sans Arabic Medio

#### Note

I caratteri utilizzati da `VectorOpenDataStandardDark` sono caratteri combinati utilizzati per la maggior parte dei glifi ma `Amazon Ember Noto Sans` per i glifi non supportati da `Amazon Ember`

Open Data Visualization Light

Open Data Visualization Light



Nome dello stile della mappa: `VectorOpenDataVisualizationLight`

Si tratta di uno stile a tema chiaro con colori tenui e un minor numero di funzioni che aiuta a comprendere i dati sovrapposti.

Questa mappa di base si basa sulla [distribuzione delle mappe OSM Daylight compilata dai contributori](#) (OSM). OpenStreetMap La comunità OSM include oltre 1,8 milioni di collaboratori che aggiornano più di 500.000 funzionalità ogni giorno. Amazon Location Service incorpora queste modifiche regolarmente.

## Fonts

Amazon Location fornisce caratteri utilizzando [GetMapGlyphs](#). Le seguenti sono le pile di caratteri disponibili per questa mappa:

- Amazon Ember Bold, Noto sans Bold
- Amazon Ember RC condensato Bold, Noto Sans Bold
- Amazon Ember RC condensato normale, Noto Sans normale
- Amazon Ember Medio, Noto Sans Medio



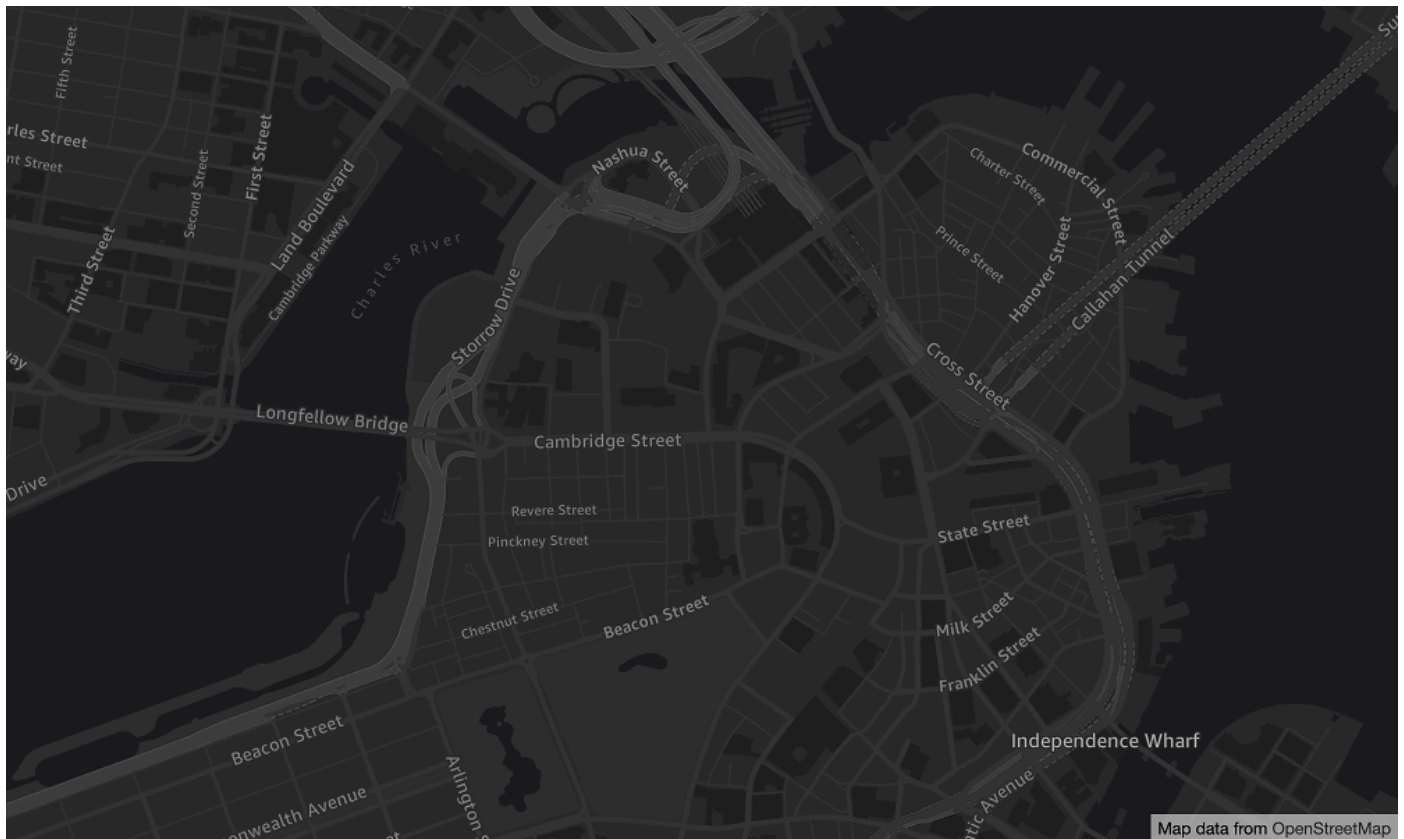
- Amazon Ember Regular Italic, Noto Sans Italic
- Amazon Ember Regolare, Noto Sans Regolare
- Amazon Ember Regolare, Noto Sans Regolare, Noto Sans Arabo Regolare
- Amazon Ember RC condensato Bold, Noto Sans Bold, Noto Sans Arabic Condensato Bold
- Amazon Ember Bold, Noto Sans Bold, Noto Sans Arabic Bold
- Amazon Ember Normale Italic, Noto Sans Italic, Noto Sans Arabo Regolare
- Amazon Ember RC condensato regolare, Noto Sans normale, Noto Sans Arabic Condensato Regolare
- Amazon Ember Medio, Noto Sans Medio, Noto Sans Arabic Medio

 Note

I caratteri utilizzati da `VectorOpenDataVisualizationLight` sono caratteri combinati utilizzati per la maggior parte dei glifi ma Amazon Ember Noto Sans per i glifi non supportati da Amazon Ember

Open Data Visualization Dark

Open Data Visualization Dark



Nome dello stile della mappa: `VectorOpenDataVisualizationDark`

Si tratta di uno stile a tema scuro con colori tenui e un minor numero di funzioni che aiuta a comprendere i dati sovrapposti.

Questa mappa di base si basa sulla [distribuzione delle mappe OSM Daylight compilata dai contributori](#) (OSM). OpenStreetMap La comunità OSM include oltre 1,8 milioni di collaboratori che aggiornano più di 500.000 funzionalità ogni giorno. Amazon Location Service incorpora queste modifiche regolarmente.

## Fonts

Amazon Location fornisce caratteri utilizzando [GetMapGlyphs](#). Le seguenti sono le pile di caratteri disponibili per questa mappa:

- Amazon Ember Bold, Noto sans Bold
- Amazon Ember RC condensato Bold, Noto Sans Bold
- Amazon Ember RC condensato normale, Noto Sans normale
- Amazon Ember Medio, Noto Sans Medio

- Amazon Ember Regular Italic, Noto Sans Italic
- Amazon Ember Regolare, Noto Sans Regolare
- Amazon Ember Regolare, Noto Sans Regolare, Noto Sans Arabo Regolare
- Amazon Ember RC condensato Bold, Noto Sans Bold, Noto Sans Arabic Condensato Bold
- Amazon Ember Bold, Noto Sans Bold, Noto Sans Arabic Bold
- Amazon Ember Normale Italic, Noto Sans Italic, Noto Sans Arabo Regolare
- Amazon Ember RC condensato regolare, Noto Sans normale, Noto Sans Arabic Condensato Regolare
- Amazon Ember Medio, Noto Sans Medio, Noto Sans Arabic Medio

#### Note

I caratteri utilizzati da `VectorOpenDataVisualizationDark` sono caratteri combinati utilizzati per la maggior parte dei glifi ma Amazon Ember Noto Sans per i glifi non supportati da Amazon Ember

## Copertura: Open Data

Open Data include mappe con copertura globale per il rendering con una [risorsa cartografica di Amazon Location Service](#).

#### Note

Gli Open Data possono essere utilizzati solo con le risorse cartografiche di Amazon Location Service. Non puoi utilizzare Open Data come fornitore di dati per supportare query di geocodifica, geocodifica inversa e ricerche o per supportare query per il calcolo di un percorso.

## Termini di utilizzo e attribuzione dei dati: Open Data

Prima di utilizzare Open Data, assicurati di poter rispettare tutti i requisiti legali applicabili, incluse le condizioni di licenza applicabili agli Open Data e AWS.

Per ulteriori informazioni sui AWS requisiti, consulta i [Termini di servizio AWS](#).

Per informazioni sulle linee guida sull'attribuzione degli Open Data, consulta OpenStreetMap le linee guida su [copyright e licenza e OpenStreetMap Licenza/Attribuzione](#).

## Segnalazione di errori e contributo agli Open Data

OpenStreetMap (OSM) e Natural Earth sono progetti di dati aperti guidati dalla comunità. Se riscontri un problema con i dati, puoi segnalare gli errori o contribuire direttamente con correzioni o suggerimenti.

- Per segnalare un errore o offrire un suggerimento in OSM, puoi creare una nota sulla mappa. Questo è un commento sulla mappa che aiuta i collaboratori ad apportare correzioni alla mappa. [Le note vengono create tramite il sito Web. OpenStreetMap](#) Per ulteriori informazioni sulle note, vedi [Notes](#) nel OpenStreetMap wiki.
- Per ulteriori informazioni su come contribuire direttamente a OpenStreetMap, inclusa l'aggiunta di posizioni e la correzione degli errori, consulta [Contribute map data](#) nel OpenStreetMap wiki.
- Per inviare una richiesta di correzione dei dati in Natural Earth, puoi inviare un problema tramite il [sito web di Natural Earth](#).

### Note

La correzione degli errori OpenStreetMap può avvenire rapidamente, tuttavia, può volerci del tempo prima che le correzioni appaiano nella distribuzione cartografica Daylight dei dati OSM utilizzata dal fornitore di Open Data. Il sito web di [Daylight Map Distribution](#) fornisce ulteriori informazioni sul processo. Inoltre, Amazon Location Service aggiorna i dati delle mappe utilizzati in Amazon Location Service circa una volta al mese.

## Funzionalità per fornitore di dati

Questa sezione descrive le funzionalità disponibili in Amazon Location Service, suddivise in categorie per fornitore di dati.

La tabella seguente fornisce una panoramica di alto livello delle funzionalità.

Fornitore di dati	Copertura geografica	Copertura delle funzionalità	Regione AWS
Esri	Globale	Mappe, luoghi, percorsi	<a href="#">Tutte le regioni</a> in cui è disponibile Amazon Location.
Afferra	<a href="#">Sud-est asiatico</a>	Mappe, luoghi, percorsi	Solo Asia Pacifico (Singapore). ap-southeast-1
QUI	Globale	Mappe, luoghi, percorsi	<a href="#">Tutte le regioni</a> in cui è disponibile Amazon Location.
Open Data (Apri dati)	Globale	Mappe	<a href="#">Tutte le regioni</a> in cui è disponibile Amazon Location.

Le seguenti schede mostrano i dettagli all'interno di ciascuna area di funzionalità.

## Map Features

La tabella seguente mostra le caratteristiche della mappa per fornitore di dati. Per ulteriori informazioni sui concetti delle mappe, vedere [Mappe](#).

Fornitore di dati	Tipi di mappe supportati	Livelli di zoom vettoriale	Livelli di zoom raster
Esri	Vettore  Raster (immagini)  Per ulteriori informazioni, consulta <a href="#">Stili di mappe Esri</a> .	0-15	0-23
Afferra	Vettore	0-14	nessuno

Fornitore di dati	Tipi di mappe supportati	Livelli di zoom vettoriale	Livelli di zoom raster
	(Solo per il <a href="#">sud-est asiatico</a> )  Per ulteriori informazioni, consulta <a href="#">Prendi gli stili delle mappe</a> .		
QUI	Vettore  Raster (immagini)  Ibrido  Per ulteriori informazioni, consulta <a href="#">Stili delle mappe HERE</a> .	1-17	0-19
Open Data (Apri dati)	Vettore  Per ulteriori informazioni, consulta <a href="#">Stili di mappa Open Data</a> .	0-15	nessuno

### Note

I livelli di zoom rappresentano le impostazioni massime e minime, come definite nelle API di ciascun provider. Aree diverse della mappa possono avere valori massimi diversi; ad esempio, le tessere oceaniche possono avere meno livelli di zoom dettagliati rispetto alle aree delle principali città.

MapLibre (e altri motori di rendering delle mappe) consentono di impostare i livelli di zoom minimi e massimi e rispettano inoltre i livelli di zoom del fornitore di dati in un'area, in modo da non dover scrivere codice per gestire queste discrepanze.

## Places and Search

La tabella seguente mostra la posizione e le funzioni di ricerca per fornitore di dati. Per ulteriori informazioni sui concetti relativi ai luoghi, vedere [Ricerca di luoghi](#).

Fornitore di dati	Geocodifica	Geocodifica inversa	Completamento automatico	GetPlace
Esri	Tutte le funzionalità, tranne:  PlaceId	Tutte le funzionalità, tranne:  TimeZone  PlaceId	Tutte le funzionalità	Tutte le funzionalità
Afferra	Tutte le funzionalità, tranne:  tipo di unità  Categorie non supportate	Tutte le funzionalità	Tutte le funzionalità	Tutte le funzionalità, tranne:  tipo di unità  SubMunicipality
QUI	Tutte le funzionalità, tranne:  numero di unità  tipo di unità  rilevanza  <a href="#">Limitazioni aggiuntive sul filtraggio</a>	Tutte le funzionalità	Tutte le funzionalità	Tutte le funzionalità, tranne:  numero di unità  tipo di unità  SubMunicipality
Open Data (Apri dati)	Non supportato	Non supportato	Non supportato	Supporta solo:

Fornitore di dati	Geocodifica	Geocodifica inversa	Completamento automatico	GetPlace
				SubMunicipality

## Route features

La tabella seguente mostra le caratteristiche del percorso per fornitore di dati. Per ulteriori informazioni sui concetti relativi alle rotte, vedere [Route](#). Per descrizioni più dettagliate delle limitazioni della matrice di percorso, vedere [Restrizioni sulle posizioni di partenza e di destinazione](#).

Fornitore di dati	Modalità di viaggio	Calcola percorso	Matrice del percorso
Esri	Auto, camion, passeggiate	Partenza e destinazioni devono trovarsi entro 400 km l'una dall'altra. Il tempo di viaggio totale non può superare i 400 minuti.  ArrivalTime non è supportato.	Fino a 10 posizioni di partenza e destinazioni.  Non supportato in Corea.  Tutte le coppie di partenza e destinazioni devono trovarsi entro 400 km l'una dall'altra.
Afferra	Auto, moto.  Passeggiate a piedi e in bicicletta in <a href="#">città selezionate</a> .	Nessun limite di distanza.	Fino a 350 posizioni di partenza e destinazione.
QUI	Auto, camion, passeggiate	Nessun limite di distanza. I percorsi che si estendono per più di 10 km	Fino a 350 posizioni di partenza e destinazione.



Fornitore di dati	Modalità di viaggio	Calcola percorso	Matrice del percorso
		all'esterno di un cerchio attorno alle posizioni di partenza e di destinazione non verranno calcolati.	Tutte le posizioni di partenza e di destinazione devono rientrare in un cerchio di 180 km.  Sono supportati percorsi più lunghi, con <a href="#">restrizioni aggiuntive</a> .
Open Data (Apri dati)	Non supportato	Non supportato	Non supportato

## Termini di utilizzo e attribuzione dei dati per i fornitori di dati

Prima di utilizzare un fornitore di dati, assicurati di poter rispettare tutti i requisiti legali applicabili, comprese le condizioni di licenza applicabili all'uso del provider.

Per ulteriori informazioni sui AWS requisiti, consulta i [Termini di servizio AWS](#).

Quando utilizzi un fornitore di dati con le tue risorse Amazon Location per la tua applicazione o documentazione, assicurati di fornire le attribuzioni per ogni fornitore di dati che utilizzi.

Per ulteriori informazioni sulla conformità e l'attribuzione per ogni fornitore di dati, consulta i seguenti argomenti.

- Esri — [Termini di utilizzo e attribuzione dei dati: Esri](#)
- Afferra — [Termini di utilizzo e attribuzione dei dati: Grab](#)
- ECCO — [Termini di utilizzo e attribuzione dei dati: QUI](#)
- Dati aperti — [Termini di utilizzo e attribuzione dei dati: Open Data](#)

## Regioni di localizzazione ed endpoint di Amazon

Amazon Location è disponibile nelle seguenti AWS regioni:

## Regioni

Nome della regione	Regione	Endpoint	Protocollo
US East (Ohio)	us-east-2	geo.us-east-2.amazonaws.com	HTTPS
US East (N. Virginia)	us-east-1	geo.us-east-1.amazonaws.com	HTTPS
US West (Oregon)	us-west-2	geo.us-west-2.amazonaws.com	HTTPS
Asia Pacifico (Mumbai)	ap-south-1	geo.ap-south-1.amazonaws.com	HTTPS
Asia Pacifico (Singapore)	ap-southeast-1	geo.ap-southeast-1.amazonaws.com	HTTPS
Asia Pacifico (Sydney)	ap-southeast-2	geo.ap-southeast-2.amazonaws.com	HTTPS
Asia Pacifico (Tokyo)	ap-northeast-1	geo.ap-northeast-1.amazonaws.com	HTTPS
Canada (Centrale)	ca-central-1	geo.ca-central-1.amazonaws.com	HTTPS

Nome della regione	Regione	Endpoint	Protocollo
Europa (Francoforte)	eu-central-1	geo.eu-central-1.amazonaws.com	HTTPS
Europa (Irlanda)	eu-west-1	geo.eu-west-1.amazonaws.com	HTTPS
Europa (Londra)	eu-west-2	geo.eu-west-2.amazonaws.com	HTTPS
Europa (Stoccolma)	eu-north-1	geo.eu-north-1.amazonaws.com	HTTPS
Sud America (São Paulo)	sa-east-1	geo.sa-east-1.amazonaws.com	HTTPS
AWS GovCloud (Stati Uniti occidentali)	us-gov-west-1	geo.us-gov-west-1.amazonaws.com	HTTPS
		geo-fips.us-gov-west-1.amazonaws.com	HTTPS

### Note

Per ulteriori informazioni su come utilizzare gli endpoint in questa tabella, vedere la sezione seguente.

## Endpoints

La sintassi generale di un endpoint regionale Amazon Location è la seguente:

```
protocol://service-code.geo.region-code.amazonaws.com
```

In questa sintassi, Amazon Location utilizza i seguenti codici di servizio:

Servizio	Codice di servizio
Amazon Location Maps	Mappe
Luoghi di Amazon Location	Luoghi
Geofence di Amazon Location	geofencing
Tracker di Amazon Location	tracking
Route di Amazon Location	Route

Ad esempio, l'endpoint regionale per Amazon Location Maps for US East (Virginia settentrionale) sarebbe: `https://maps.geo.us-east-1.amazonaws.com`.

## Endpoint operativi tramite API

La sintassi per un endpoint del piano di controllo di Amazon Location Service è la seguente:

```
protocol://cp.service-code.geo.region-code.amazonaws.com
```

Le azioni del piano di controllo per Amazon Location Service sono:

Servizio	Endpoint	Operazione API
Amazon Location Maps	<code>https://cp.maps.geo. <i>regione</i>.amazonaw s.com</code>	<a href="#">CreateMap</a> <a href="#">DeleteMap</a> <a href="#">DescribeMap</a>

Servizio	Endpoint	Operazione API
		<a href="#">ListMaps</a> <a href="#">UpdateMap</a>
Luoghi di Amazon Location	<a href="https://cp.places.geo.&lt;i&gt;regione&lt;/i&gt;.amazonaws.com">https://cp.places.geo.<i>regione</i>.amazonaws.com</a>	<a href="#">CreatePlaceIndex</a> <a href="#">DeletePlaceIndex</a> <a href="#">DescribePlaceIndex</a> <a href="#">ListPlaceIndexes</a> <a href="#">UpdatePlaceIndex</a>
Geofence di Amazon Location	<a href="https://cp.geofencing.geo.&lt;i&gt;regione&lt;/i&gt;.amazonaws.com">https://cp.geofencing.geo.<i>regione</i>.amazonaws.com</a>	<a href="#">CreateGeofenceCollection</a> <a href="#">DeleteGeofenceCollection</a> <a href="#">DescribeGeofenceCollection</a> <a href="#">ListGeofenceCollections</a> <a href="#">UpdateGeofenceCollection</a>
Tracker di Amazon Location	<a href="https://cp.tracking.geo.&lt;i&gt;regione&lt;/i&gt;.amazonaws.com">https://cp.tracking.geo.<i>regione</i>.amazonaws.com</a>	<a href="#">CreateTracker</a> <a href="#">DeleteTracker</a> <a href="#">DescribeTracker</a> <a href="#">UpdateTracker</a> <a href="#">ListTrackers</a> <a href="#">AssociateTrackerConsumer</a> <a href="#">DisassociateTrackerConsumer</a> <a href="#">ListTrackerConsumers</a>

Servizio	Endpoint	Operazione API
Route di Amazon Location	<code>https://cp.routes.geo. <i>regione</i> .amazonaw s.com</code>	<a href="#">CreateRouteCalculator</a> <a href="#">DeleteRouteCalculator</a> <a href="#">DescribeRouteCalculator</a> <a href="#">ListRouteCalculators</a> <a href="#">UpdateRouteCalculator</a>
Metadati sulla posizione di Amazon	<code>https://cp.metadata.geo. <i>regione</i> .amazonaw s.com</code>	<a href="#">CreateKey</a> <a href="#">DeleteKey</a> <a href="#">DescribeKey</a> <a href="#">ListKeys</a> <a href="#">UpdateKey</a>

## Quote di Amazon Location Service

Questo argomento fornisce un riepilogo dei limiti di tariffa e delle quote per Amazon Location Service.

### Note

Se hai bisogno di una quota più alta, puoi utilizzare la console Service Quotas per [richiedere aumenti delle quote per le quote](#) regolabili. Quando richiedi un aumento della quota, seleziona la regione in cui desideri aumentare la quota poiché la maggior parte delle quote è specifica della regione. AWS

Le Service Quotas sono il numero massimo di risorse che puoi avere a disposizione per AWS account e AWS regione. Amazon Location Service rifiuta le richieste aggiuntive che superano la quota di servizio.

Limiti di tariffa (quote che iniziano con Rate of... ) sono il numero massimo di richieste al secondo, con una frequenza di burst pari all'80% del limite in qualsiasi parte del secondo, definita per ogni

operazione API. Amazon Location Service limita le richieste che superano il limite di velocità dell'operazione.

Nome	Predefinita	Adattata	Descrizione
Risorse della chiave API per account	Ogni regione supportata: 500	No	Il numero massimo di risorse chiave API (attive o scadute) che puoi avere per account.
Risorse di raccolta di geofence per account	Ogni regione supportata: 1.500	<a href="#">Sì</a>	Il numero massimo di risorse di raccolta geofence che puoi creare per account.
Geofence per raccolta geofence	Ogni regione supportata: 50.000	No	Il numero massimo di geofence che puoi creare per ogni raccolta geofence.
Risorse di mappa per account	Ogni regione supportata: 40	<a href="#">Sì</a>	Il numero massimo di risorse Map che puoi creare per account.
Risorse dell'indice di posizionamento per account	Ogni regione supportata: 40	<a href="#">Sì</a>	Il numero massimo di risorse dell'indice di posizione che puoi creare per account.
Frequenza delle richieste Associate TrackerConsumer API	Ogni regione supportata: 10 al secondo	<a href="#">Sì</a>	Il numero massimo di AssociateTrackerConsumer richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della

Nome	Predefinita	Adatta	Descrizione
			larghezza di banda della rete).
Frequenza delle richieste BatchDeleteDevicePositionHistory API	Ogni regione supportata: 50 al secondo	<a href="#">Sì</a>	Il numero massimo di BatchDeleteDevicePositionHistory richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).
Frequenza delle richieste BatchDeleteGeofence API	Ogni regione supportata: 50 al secondo	<a href="#">Sì</a>	Il numero massimo di BatchDeleteGeofence richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).
Frequenza delle richieste BatchEvaluateGeofences API	Ogni regione supportata: 50 al secondo	<a href="#">Sì</a>	Il numero massimo di BatchEvaluateGeofences richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).



Nome	Predefinita	Adatta e	Descrizione
Frequenza delle richieste BatchGetDevicePosition API	Ogni regione supportata: 50 al secondo	<a href="#">Sì</a>	Il numero massimo di BatchGetDevicePosition richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).
Frequenza delle richieste BatchPutGeofence API	Ogni regione supportata: 50 al secondo	<a href="#">Sì</a>	Il numero massimo di BatchPutGeofence richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).
Frequenza delle richieste BatchUpdateDevicePosition API	Ogni regione supportata: 50 al secondo	<a href="#">Sì</a>	Il numero massimo di BatchUpdateDevicePosition richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).

Nome	Predefinita	Adattata	Descrizione
Frequenza delle richieste CalculateRoute API	Ogni regione supportata: 10 al secondo	<a href="#">Sì</a>	Il numero massimo di CalculateRoute richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).
Frequenza delle richieste CalculateRouteMatrix API	Ogni regione supportata: 5 al secondo	<a href="#">Sì</a>	Il numero massimo di CalculateRouteMatrix richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte e a limitazione (della larghezza di banda della rete).
Frequenza delle richieste CreateGeofenceCollection API	Ogni regione supportata: 10 al secondo	<a href="#">Sì</a>	Il numero massimo di CreateGeofenceCollection richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte e a limitazione (della larghezza di banda della rete).

Nome	Predefinita	Adattate	Descrizione
Frequenza delle richieste CreateKey API	Ogni regione supportata: 10 al secondo	<a href="#">Sì</a>	Il numero massimo di CreateKey richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).
Frequenza delle richieste CreateMap API	Ogni regione supportata: 10 al secondo	<a href="#">Sì</a>	Il numero massimo di CreateMap richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).
Frequenza delle richieste CreatePlaceIndex API	Ogni regione supportata: 10 al secondo	<a href="#">Sì</a>	Il numero massimo di CreatePlaceIndex richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte e a limitazione (della larghezza di banda della rete).

Nome	Predefinita	Adattata	Descrizione
Frequenza delle richieste CreateRouteCalculator API	Ogni regione supportata: 10 al secondo	<a href="#">Sì</a>	Il numero massimo di CreateRouteCalculator richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).
Frequenza delle richieste CreateTracker API	Ogni regione supportata: 10 al secondo	<a href="#">Sì</a>	Il numero massimo di CreateTracker richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).
Frequenza delle richieste DeleteGeofenceCollection API	Ogni regione supportata: 10 al secondo	<a href="#">Sì</a>	Il numero massimo di DeleteGeofenceCollection richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).

Nome	Predefinita	Adattate	Descrizione
Frequenza delle richieste DeleteKey API	Ogni regione supportata: 10 al secondo	<a href="#">Sì</a>	Il numero massimo di DeleteKey richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).
Frequenza delle richieste DeleteMap API	Ogni regione supportata: 10 al secondo	<a href="#">Sì</a>	Il numero massimo di DeleteMap richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).
Frequenza delle richieste DeletePlaceIndex API	Ogni regione supportata: 10 al secondo	<a href="#">Sì</a>	Il numero massimo di DeletePlaceIndex richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte e a limitazione (della larghezza di banda della rete).

Nome	Predefinita	Adattate	Descrizione
Frequenza delle richieste DeleteRouteCalculator API	Ogni regione supportata: 10 al secondo	<a href="#">Sì</a>	Il numero massimo di DeleteRouteCalculator richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).
Frequenza delle richieste DeleteTracker API	Ogni regione supportata: 10 al secondo	<a href="#">Sì</a>	Il numero massimo di DeleteTracker richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).
Frequenza delle richieste DescribeGeofenceCollection API	Ogni regione supportata: 10 al secondo	<a href="#">Sì</a>	Il numero massimo di DescribeGeofenceCollection richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).

Nome	Predefinita	Adattate	Descrizione
Frequenza delle richieste DescribeKey API	Ogni regione supportata: 10 al secondo	<a href="#">Sì</a>	Il numero massimo di DescribeKey richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).
Frequenza delle richieste DescribeMap API	Ogni regione supportata: 10 al secondo	<a href="#">Sì</a>	Il numero massimo di DescribeMap richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).
Frequenza delle richieste DescribePlaceIndex API	Ogni regione supportata: 10 al secondo	<a href="#">Sì</a>	Il numero massimo di DescribePlaceIndex richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte e a limitazione (della larghezza di banda della rete).

Nome	Predefinita	Adattata	Descrizione
Frequenza delle richieste DescribeRouteCalculator API	Ogni regione supportata: 10 al secondo	<a href="#">Sì</a>	Il numero massimo di DescribeRouteCalculator richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).
Frequenza delle richieste DescribeTracker API	Ogni regione supportata: 10 al secondo	<a href="#">Sì</a>	Il numero massimo di DescribeTracker richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).
Frequenza delle richieste DisassociateTrackerConsumer API	Ogni regione supportata: 10 al secondo	<a href="#">Sì</a>	Il numero massimo di DisassociateTrackerConsumer richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).



Nome	Predefinita	Adatta e	Descrizione
Frequenza delle richieste ForecastGeofenceEvents API	Ogni regione supportata: 50 al secondo	<a href="#">Sì</a>	Il numero massimo di ForecastGeofenceEvents richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).
Frequenza delle richieste GetDevicePosition API	Ogni regione supportata: 50 al secondo	<a href="#">Sì</a>	Il numero massimo di GetDevicePosition richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).
Frequenza delle richieste GetDevicePositionHistory API	Ogni regione supportata: 50 al secondo	<a href="#">Sì</a>	Il numero massimo di GetDevicePositionHistory richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).

Nome	Predefinita	Adatta e	Descrizione
Frequenza delle richieste GetGeofence API	Ogni regione supportata: 50 al secondo	<a href="#">Sì</a>	Il numero massimo di GetGeofence richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).
Frequenza delle richieste GetMapGlyphs API	Ogni regione supportata: 50 al secondo	<a href="#">Sì</a>	Il numero massimo di GetMapGlyphs richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).
Frequenza delle richieste GetMapSprites API	Ogni regione supportata: 50 al secondo	<a href="#">Sì</a>	Il numero massimo di GetMapSprites richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).

Nome	Predefinita	Adatta e	Descrizione
Frequenza delle richieste GetMapStyleDescriptor API	Ogni regione supportata: 50 al secondo	<a href="#">Sì</a>	Il numero massimo di GetMapStyleDescriptor richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).
Frequenza delle richieste GetMapTile API	Ogni regione supportata: 500 al secondo	<a href="#">Sì</a>	Il numero massimo di GetMapTile richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).
Frequenza delle richieste GetPlace API	Ogni regione supportata: 50 al secondo	<a href="#">Sì</a>	Il numero massimo di GetPlace richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).

Nome	Predefinita	Adatta	Descrizione
Frequenza delle richieste ListDevicePositions API	Ogni regione supportata: 50 al secondo	<a href="#">Sì</a>	Il numero massimo di ListDevicePositions richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).
Frequenza delle richieste ListGeofenceCollections API	Ogni regione supportata: 10 al secondo	<a href="#">Sì</a>	Il numero massimo di ListGeofenceCollections richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).
Frequenza delle richieste ListGeofences API	Ogni regione supportata: 50 al secondo	<a href="#">Sì</a>	Il numero massimo di ListGeofences richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).

Nome	Predefinita	Adattata	Descrizione
Frequenza delle richieste ListKeys API	Ogni regione supportata: 10 al secondo	<a href="#">Sì</a>	Il numero massimo di ListKeys richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).
Frequenza delle richieste ListMaps API	Ogni regione supportata: 10 al secondo	<a href="#">Sì</a>	Il numero massimo di ListMaps richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).
Frequenza delle richieste ListPlace Indexes API	Ogni regione supportata: 10 al secondo	<a href="#">Sì</a>	Il numero massimo di ListPlaceIndexes richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).

Nome	Predefinita	Adatta e	Descrizione
Frequenza delle richieste ListRouteCalculators API	Ogni regione supportata: 10 al secondo	<a href="#">Sì</a>	Il numero massimo di ListRouteCalculators richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).
Frequenza delle richieste ListTagsForResource API	Ogni regione supportata: 10 al secondo	<a href="#">Sì</a>	Il numero massimo di ListTagsForResource richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).
Frequenza delle richieste ListTrackerConsumers API	Ogni regione supportata: 10 al secondo	<a href="#">Sì</a>	Il numero massimo di ListTrackerConsumers richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).

Nome	Predefinita	Adatta	Descrizione
Frequenza delle richieste ListTrackers API	Ogni regione supportata: 10 al secondo	<a href="#">Sì</a>	Il numero massimo di ListTrackers richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).
Frequenza delle richieste PutGeofence API	Ogni regione supportata: 50 al secondo	<a href="#">Sì</a>	Il numero massimo di PutGeofence richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).
Frequenza delle richieste SearchPlaceIndexForPosition API	Ogni regione supportata: 50 al secondo	<a href="#">Sì</a>	Il numero massimo di SearchPlaceIndexForPosition richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).

Nome	Predefinita	Adatta e	Descrizione
Frequenza delle richieste SearchPlaceIndexForSuggestions API	Ogni regione supportata: 50 al secondo	<a href="#">Sì</a>	Il numero massimo di SearchPlaceIndexForSuggestions richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).
Frequenza delle richieste SearchPlaceIndexForText API	Ogni regione supportata: 50 al secondo	<a href="#">Sì</a>	Il numero massimo di SearchPlaceIndexForText richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).
Frequenza delle richieste TagResource API	Ogni regione supportata: 10 al secondo	<a href="#">Sì</a>	Il numero massimo di TagResource richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).



Nome	Predefinita	Adatta e	Descrizione
Frequenza delle richieste UntagResource API	Ogni regione supportata: 10 al secondo	<a href="#">Sì</a>	Il numero massimo di UntagResource richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).
Frequenza delle richieste UpdateGeofenceCollection API	Ogni regione supportata: 10 al secondo	<a href="#">Sì</a>	Il numero massimo di UpdateGeofenceCollection richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).
Frequenza delle richieste UpdateKey API	Ogni regione supportata: 10 al secondo	<a href="#">Sì</a>	Il numero massimo di UpdateKey richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).

Nome	Predefinita	Adattata	Descrizione
Frequenza delle richieste UpdateMap API	Ogni regione supportata: 10 al secondo	<a href="#">Sì</a>	Il numero massimo di UpdateMap richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).
Frequenza delle richieste UpdatePlaceIndex API	Ogni regione supportata: 10 al secondo	<a href="#">Sì</a>	Il numero massimo di UpdatePlaceIndex richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte e a limitazione (della larghezza di banda della rete).
Frequenza delle richieste UpdateRouteCalculator API	Ogni regione supportata: 10 al secondo	<a href="#">Sì</a>	Il numero massimo di UpdateRouteCalculator richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte e a limitazione (della larghezza di banda della rete).

Nome	Predefinita	Adattate	Descrizione
Frequenza delle richieste UpdateTracker API	Ogni regione supportata: 10 al secondo	<a href="#">Sì</a>	Il numero massimo di UpdateTracker richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).
Frequenza delle richieste VerifyDevicePosition API	Ogni regione supportata: 50 al secondo	<a href="#">Sì</a>	Il numero massimo di VerifyDevicePosition richieste che puoi effettuare al secondo. Ulteriori richieste verranno sottoposte a limitazione (della larghezza di banda della rete).
Risorse di Route Calculator per account	Ogni regione supportata: 40	<a href="#">Sì</a>	Il numero massimo di risorse del calcolatore di route che puoi creare per account.
Consumatori tracker per tracker	Ogni Regione supportata: 5	No	Il numero massimo di raccolte geofence a cui puoi associare la risorsa del tracker.
Risorse del tracker per account	Ogni regione supportata: 500	<a href="#">Sì</a>	Il numero massimo di risorse tracker che puoi creare per account.

### Note

Puoi monitorare l'utilizzo rispetto alle tue quote con Cloudwatch. Per ulteriori informazioni, consulta [Utilizzato CloudWatch per monitorare l'utilizzo rispetto alle quote](#).

## Gestione delle quote del servizio Amazon Location

Amazon Location Service è integrato con Service Quotas, un AWS servizio che ti consente di visualizzare e gestire le tue quote da una posizione centrale. Per ulteriori informazioni, consulta [Cos'è Service Quotas?](#) nella Guida per l'utente di Service Quotas.

Service Quotas semplifica la ricerca del valore delle quote del servizio Amazon Location.

### AWS Management Console

Per visualizzare le quote del servizio Amazon Location utilizzando la console

1. Apri la console Service Quotas all'indirizzo <https://console.aws.amazon.com/servicequotas/>
2. Nel pannello di navigazione, scegli Servizi AWS .
3. Dall'elenco dei AWS servizi, cerca e seleziona Amazon Location.

Nell'elenco Service Quotas, è possibile visualizzare il nome della quota di servizio, il valore applicato (se è disponibile), la quota predefinita AWS e se il valore della quota è adattabile.

4. Per visualizzare ulteriori informazioni su una quota di servizio, ad esempio la descrizione, scegli il nome della quota.
5. (Facoltativo) Per richiedere un aumento della quota, selezionare la quota che si desidera aumentare, selezionare Richiedi aumento quota, immettere o selezionare le informazioni richieste e selezionare Richiedi.

Per maggiori informazioni sulle quote di servizio utilizzando la console, consulta [Guida per l'utente di Service Quotas](#). Per richiedere un aumento delle quote, consulta [Richiesta di aumento delle quote](#) nella Guida per l'utente di Service Quotas.

### AWS CLI

Per visualizzare le quote del servizio Amazon Location utilizzando AWS CLI

Esegui il comando seguente per visualizzare le quote predefinite di Amazon Location.

```
aws service-quotas list-aws-default-service-quotas \
  --query 'Quotas[*]'.
{Adjustable:Adjustable,Name:QuotaName,Value:Value,Code:QuotaCode}' \
  --service-code geo \
  --output table
```

Per utilizzare meglio le quote di servizio utilizzando il AWS CLI, vedere Service Quotas [Command AWS CLI Reference](#). Per richiedere un aumento delle quote, consultare il comando [request-service-quota-increase](#) nella [Documentazione di riferimento sui comandi AWS CLI](#).

# Guida introduttiva a utilizzare Amazon Location Service come sviluppatore

Puoi utilizzare Amazon Location Service per fornire funzionalità relative alla geografia per app in diversi fattori di forma e sistemi, inclusi servizi Web di backend, applicazioni Web e applicazioni mobili. Sono disponibili molti strumenti per aiutarti a creare le tue applicazioni, tra cui SDK, librerie e codice di esempio.

Questa sezione fornisce informazioni e link per aiutarti a iniziare a usare Amazon Location. In particolare, i seguenti argomenti forniscono le informazioni che possono esserti più utili:

- [Scenari e casi d'uso](#)— Un elenco di scenari di sviluppo e di come Amazon Location Service può aiutarti a completarli.
- [SDK e strumenti Amazon Location](#): kit di sviluppo software (SDK) e librerie che ti aiuteranno nella programmazione con Amazon Location.
- [Riferimento all'API Amazon Location Service](#): un riferimento alle principali API di Amazon Location fornite con l' AWS SDK.
- [Esempi di codice](#): questa sezione fornisce esempi che ti aiuteranno a iniziare o ad aggiungere funzionalità alla tua applicazione esistente.
- [Tutorial di avvio rapido](#): questo tutorial mostra come creare la prima applicazione. Esistono versioni del tutorial per creare un'applicazione web o un'applicazione mobile basata su Android.
- [Concetti di Amazon Location Service](#)— Questa sezione di questa guida descrive i concetti di base di Amazon Location, incluse le sezioni su Maps, Places search, Routes, Geofences and Tracker.
- [Amplify](#): Amplify è una soluzione completa che racchiude gran parte delle funzionalità necessarie per creare applicazioni Web e mobili utilizzando il. Cloud AWS Se stai già utilizzando Amplify, o scegli di utilizzare Amplify, dispone di una libreria geografica che utilizza Amazon Location Service integrata che puoi utilizzare. [Per iniziare a usare Amplify Geo, consulta la documentazione qui.](#)

## Scenari e casi d'uso

Amazon Location Service è un servizio che viene eseguito in Cloud AWS. Puoi richiamarlo dalle tue istanze Amazon EC2 nel cloud, ma molte applicazioni di mappatura verranno eseguite su dispositivi o su una combinazione di dispositivi e cloud. Di seguito sono elencati solo alcuni scenari tipici e il modo in cui potresti affrontarli al loro sviluppo.

- Un'applicazione di backend che ti aiuta a ottimizzare i percorsi per i conducenti della tua flotta.

Puoi scrivere un'applicazione da eseguire su [Amazon EC2](#) Cloud AWS che utilizza Amazon Location Service per [calcolare le matrici di percorso](#) come input per un ottimizzatore di rotte per la tua flotta. Usa l'[AWS SDK](#) per effettuare chiamate verso Amazon Location.

- Un'applicazione web che consente ai tuoi clienti di trovare le sedi della tua attività.

Puoi creare un sito Web eseguibile su istanze Amazon EC2, inclusa un'applicazione basata sulla posizione. [Utilizza l'AWS SDK per sviluppare un'applicazione web JavaScript per cercare le posizioni utilizzando la ricerca di luoghi e visualizzare i risultati su una mappa utilizzando](#) MapLibre Usa Amazon Location SDK per semplificare la programmazione con la posizione.

- Aggiungi funzionalità di localizzazione a un'applicazione iOS o Android esistente.

Puoi utilizzare l' AWS SDK per Swift (iOS) o [Kotlin](#) (Android) per effettuare chiamate ad Amazon Location e aggiungere funzionalità di [ricerca di luoghi](#) e [mappe](#) alla tua applicazione. Utilizzalo per MapLibre renderizzare le mappe. Sono disponibili [AWS SDK](#) aggiuntivi per altre lingue.

- Tieni traccia delle risorse (dispositivi o veicoli) e ricevi aggiornamenti quando entrano o escono da aree definite da te.

Un'applicazione per tracciare i dispositivi è composta da diverse parti.

- Per ogni dispositivo che state monitorando deve essere creata una risorsa di [tracciamento](#) per tracciarlo. Deve inviare aggiornamenti di posizione ad Amazon Location Service, ad esempio utilizzando [MQTT](#).
- Crea [geofence](#) per definire le aree in cui desideri ricevere gli eventi di entrata e uscita dei tuoi asset.
- Puoi usare [Amazon EC2](#) o rispondere [AWS Lambda](#) ai tuoi eventi quando gli asset entrano o escono dalle aree di geofence.
- Puoi ampliare questa funzionalità per creare applicazioni web o per dispositivi per tracciare e visualizzare le posizioni delle risorse su mappe.

La sezione seguente fornisce dettagli sugli strumenti e le librerie disponibili per l'uso con ogni aspetto di Amazon Location Service.

## SDK e strumenti per l'utilizzo di Amazon Location Service

Esistono diversi strumenti che ti aiuteranno a utilizzare Amazon Location Service.

- **AWS SDK:** i kit di sviluppo AWS software (SDK) sono disponibili in molti linguaggi di programmazione più diffusi e forniscono un'API, esempi di codice e documentazione che semplifica la creazione di applicazioni nel linguaggio preferito. Gli AWS SDK includono le API e le funzionalità principali di Amazon Location, incluso l'accesso a Maps, Places search, Routes, Geofence e Trackers. Per ulteriori informazioni sugli SDK disponibili per l'uso con Amazon Location Service per diverse applicazioni e lingue, consulta [SDK per lingua](#).
- **MapLibre**— Amazon Location Service consiglia di eseguire il rendering delle mappe utilizzando il motore di [MapLibre](#) rendering. MapLibre è un motore per la visualizzazione di mappe in applicazioni web o mobili. MapLibre ha anche un modello di plugin e supporta l'interfaccia utente per la ricerca e i percorsi in alcune lingue e piattaforme. Per ulteriori informazioni sull'utilizzo MapLibre e sulle funzionalità che offre, consulta [MapLibre](#).
- **Amazon Location SDK:** Amazon Location SDK è un set di librerie open source che semplificano lo sviluppo di applicazioni con Amazon Location Service. Le librerie forniscono funzionalità per supportare l'autenticazione per applicazioni mobili e Web, il tracciamento della posizione per le applicazioni mobili, la conversione tra tipi di dati Amazon Location e [GeoJSON](#), nonché un pacchetto ospitato del client Amazon Location per AWS l'SDK v3. Per ulteriori informazioni su Amazon Location SDK, consulta [SDK di localizzazione Amazon](#).

## SDK per lingua

Le tabelle seguenti forniscono informazioni sugli AWS SDK e sulle MapLibre versioni per linguaggi e framework, per tipo di applicazione: applicazione web, mobile o backend.

### Versioni SDK

Ti consigliamo di utilizzare la build più recente dell' AWS SDK e qualsiasi altro SDK che utilizzi nei tuoi progetti e di mantenere gli SDK aggiornati. L' AWS SDK offre le caratteristiche e le funzionalità più recenti e anche aggiornamenti di sicurezza. Per trovare la versione più recente dell' AWS SDK per JavaScript, ad esempio, consulta [l'argomento sull'installazione del browser](#) nell'AWS SDK per la documentazione. JavaScript

## Web frontend

I seguenti AWS SDK e MapLibre versioni sono disponibili per lo sviluppo di applicazioni web frontend.



Linguaggio/Framework	SDK AWS	Framework di rendering
Completamente supportato		
JavaScript	<a href="https://aws.amazon.com/sdk-for-javascript/">https://aws.amazon.com/sdk-for-javascript/</a>	<a href="https://maplibre.org/projects/maplibre-gl-js/">https://maplibre.org/projects/maplibre-gl-js/</a>
ReactJS	<a href="https://aws.amazon.com/sdk-for-javascript">https://aws.amazon.com/sdk-for-javascript</a>	<a href="https://github.com/maplibre/maplibre-react-native">https://github.com/maplibre/maplibre-react-native</a>
TypeScript	<a href="https://aws.amazon.com/sdk-for-javascript/">https://aws.amazon.com/sdk-for-javascript/</a>	<a href="https://maplibre.org/projects/maplibre-gl-js/">https://maplibre.org/projects/maplibre-gl-js/</a>
Parzialmente supportato		
Flutter	<a href="https://docs.amplify.aws/start/q/integration/flutter/">https://docs.amplify.aws/start/q/integration/flutter/</a>  Flutter non è ancora completamente supportato da AWS, ma il supporto offerto tramite Amplify è limitato.	<a href="https://github.com/maplibre/flutter-maplibre-gl">https://github.com/maplibre/flutter-maplibre-gl</a>  La libreria MapLibre Flutter è considerata sperimentale.
Node.js	<a href="https://aws.amazon.com/sdk-for-javascript">https://aws.amazon.com/sdk-for-javascript</a>	Non è disponibile alcun MapLibre supporto per Node.js.
PHP	<a href="https://aws.amazon.com/sdk-for-php/">https://aws.amazon.com/sdk-for-php/</a>	Non è disponibile alcun MapLibre supporto per PHP.

## Mobile frontend

I seguenti AWS SDK e MapLibre versioni sono disponibili per lo sviluppo di applicazioni frontend per dispositivi mobili.

Linguaggio/Framework	AWS SDK	Framework di rendering
Completamente supportato		
Java	<a href="https://aws.amazon.com/sdk-for-java/">https://aws.amazon.com/sdk-for-java/</a>	<a href="https://maplibre.org/projects/maplibre-native/">https://maplibre.org/projects/maplibre-native/</a>
Kotlin	<p><a href="https://aws.amazon.com/sdk-for-kotlin/">https://aws.amazon.com/sdk-for-kotlin/</a></p> <p>SDK di autenticazione mobile Amazon Location Service per Android: <a href="https://github.com/aws-geospatial/-sdk-android">amazon-location-mobile-auth</a> <a href="https://github.com/aws-geospatial/-sdk-android">https://github.com/aws-geospatial/-sdk-android</a></p> <p>SDK di localizzazione mobile di Amazon Location Service per Android: <a href="https://github.com/aws-geospatial/-sdk-android">amazon-location-mobile-tracking</a> <a href="https://github.com/aws-geospatial/-sdk-android">https://github.com/aws-geospatial/-sdk-android</a></p>	<p><a href="https://maplibre.org/projects/maplibre-native/">https://maplibre.org/projects/maplibre-native/</a></p> <p>Richiede associazioni personalizzate, in quanto MapLibre è basato su Java.</p>
ObjectiveC	<a href="https://github.com/aws-amplify/aws-sdk-ios">https://github.com/aws-amplify/aws-sdk-ios</a>	<a href="https://maplibre.org/projects/maplibre-native/">https://maplibre.org/projects/maplibre-native/</a>
ReactNative	<a href="https://aws.amazon.com/sdk-for-javascript/">https://aws.amazon.com/sdk-for-javascript/</a>	<a href="https://github.com/maplibre/maplibre-react-native">https://github.com/maplibre/maplibre-react-native</a>
Swift	<p><a href="https://aws.amazon.com/sdk-for-swift/">https://aws.amazon.com/sdk-for-swift/</a></p> <p>SDK di autenticazione mobile Amazon Location Service per iOS: <a href="https://github.com/aws-geospatial/-sdk-ios">amazon-location-mobile-auth</a> <a href="https://github.com/aws-geospatial/-sdk-ios">https://github.com/aws-geospatial/-sdk-ios</a></p>	<a href="https://maplibre.org/projects/maplibre-native/">https://maplibre.org/projects/maplibre-native/</a>

Linguaggio/Framework	AWS SDK	Framework di rendering
	SDK di monitoraggio mobile Amazon Location Service per iOS: <a href="https://github.com/aws-geospatial/-sdk-ios">amazon-location-mobile-tracking</a> <a href="https://github.com/aws-geospatial/-sdk-ios">https://github.com/aws-geospatial/ -sdk-ios</a>	
Parzialmente supportato		
Flutter	<a href="https://docs.amplify.aws/start/q/integration/flutter/">https://docs.amplify.aws/start/q/integration/flutter/</a>  Flutter non è ancora completamente supportato da AWS, ma il supporto offerto tramite Amplify è limitato.	<a href="https://github.com/maplibre/flutter-maplibre-gl">https://github.com/maplibre/flutter-maplibre-gl</a>  La libreria MapLibre Flutter è considerata sperimentale.

## Backend application

I seguenti AWS SDK sono disponibili per lo sviluppo di applicazioni di backend. MapLibre non è elencato qui, perché il rendering delle mappe non è in genere necessario per le applicazioni di backend.

Lingua	SDK AWS
.NET	<a href="https://aws.amazon.com/sdk-for-net">https://aws.amazon.com/sdk-for-net</a>
C++	<a href="https://aws.amazon.com/sdk-for-cpp/">https://aws.amazon.com/sdk-for-cpp/</a>
Go	<a href="https://aws.amazon.com/sdk-for-go/">https://aws.amazon.com/sdk-for-go/</a>
Java	<a href="https://aws.amazon.com/sdk-for-java/">https://aws.amazon.com/sdk-for-java/</a>
JavaScript	<a href="https://aws.amazon.com/sdk-for-javascript/">https://aws.amazon.com/sdk-for-javascript/</a>
Node.js	<a href="https://aws.amazon.com/sdk-for-javascript/">https://aws.amazon.com/sdk-for-javascript/</a>

Lingua	SDK AWS
TypeScript	<a href="https://aws.amazon.com/sdk-for-javascript/">https://aws.amazon.com/sdk-for-javascript/</a>
Kotlin	<a href="https://aws.amazon.com/sdk-for-kotlin/">https://aws.amazon.com/sdk-for-kotlin/</a>
PHP	<a href="https://aws.amazon.com/sdk-for-php/">https://aws.amazon.com/sdk-for-php/</a>
Python	<a href="https://aws.amazon.com/sdk-for-python/">https://aws.amazon.com/sdk-for-python/</a>
Ruby	<a href="https://aws.amazon.com/sdk-for-ruby/">https://aws.amazon.com/sdk-for-ruby/</a>
Rust	<a href="https://aws.amazon.com/sdk-for-rust/">https://aws.amazon.com/sdk-for-rust/</a>  L' AWS SDK per Rust è disponibile in anteprima per sviluppatori.

## Utilizzo di MapLibre strumenti e librerie con Amazon Location

Uno degli strumenti più importanti per creare applicazioni interattive con Amazon Location è MapLibre. [MapLibre](#) è principalmente un motore di rendering per la visualizzazione di mappe in un'applicazione web o mobile. Tuttavia, include anche il supporto per i plug-in e fornisce funzionalità per lavorare con altri aspetti di Amazon Location. Di seguito vengono descritti gli strumenti che puoi utilizzare, in base all'area di localizzazione con cui desideri lavorare.

### Note

Per utilizzare qualsiasi aspetto di Amazon Location, installa l'[AWS SDK per la lingua che desideri utilizzare](#).

- Mappe

Per visualizzare le mappe nella tua applicazione, è necessario un motore di rendering delle mappe che utilizzi i dati forniti da Amazon Location e disegni sullo schermo. I motori di rendering delle mappe forniscono anche funzionalità per ingrandire e ingrandire la mappa o per aggiungere marcatori, puntine e altre annotazioni alla mappa.

Amazon Location Service consiglia di eseguire il rendering delle mappe utilizzando il motore di [MapLibre](#) rendering. MapLibre GL JS è un motore per la visualizzazione di mappe JavaScript, mentre MapLibre Native fornisce mappe per iOS o Android.

MapLibre dispone anche di un ecosistema di plug-in per estendere le funzionalità di base. Per maggiori informazioni, visita <https://maplibre.org/maplibre-gl-js-docs/plugins/>.

- Ricerca luoghi

Per semplificare la creazione di un'interfaccia utente di ricerca, puoi utilizzare il [MapLibre geocodificatore](#) per il web (le applicazioni Android possono utilizzare il plug-in [Android Places](#)).

Usa la [libreria di geocodificatori Amazon Location for Maplibre](#) per semplificare il processo di utilizzo di Amazon Location nelle applicazioni. `amazon-location-for-maplibre-gl-geocoder` JavaScript

- Percorsi

Per visualizzare i percorsi sulla mappa, usa [MapLibre indicazioni stradali](#).

- Geofence e tracker

MapLibre non dispone di rendering o strumenti specifici per i geofence e il tracciamento, ma puoi utilizzare la funzionalità di rendering e i [plug-in](#) per mostrare i geofence e i dispositivi tracciati sulla mappa.

I dispositivi monitorati possono utilizzare [MQTT](#) o inviare manualmente gli aggiornamenti ad Amazon Location Service. È possibile rispondere agli eventi Geofence utilizzando [AWS Lambda](#)

Sono disponibili molte librerie open source per fornire funzionalità aggiuntive per Amazon Location Service, ad esempio [Turf](#) che fornisce funzionalità di analisi spaziale.

Molte librerie utilizzano dati in formato [GeoJSON](#) standard aperto. Amazon Location Service fornisce una libreria per supportare l'utilizzo di GeoJSON nelle applicazioni JavaScript. Per ulteriori informazioni, consulta la sezione successiva [Amazon Location SDK e librerie](#).

## Plugin Amazon Location MapLibre Geocoder

Il plug-in di MapLibre geocodifica Amazon Location è progettato per semplificare l'integrazione della funzionalità di Amazon Location nelle tue JavaScript applicazioni, quando lavori con il rendering di mappe e la geocodifica utilizzando la libreria [maplibre-gl-geocoder](#)

## Installazione

Puoi installare il plug-in Amazon Location MapLibre geocoder da NPM per l'utilizzo con i moduli, con questo comando:

```
npm install @aws/amazon-location-for-maplibre-gl-geocoder
```

Puoi importarlo in un file HTML per utilizzarlo direttamente nel browser, con uno script:

```
<script src="https://www.unpkg.com/@aws/amazon-location-for-maplibre-gl-geocoder@1">/script<
```

## Utilizzo con il modulo

Questo codice configura una mappa Maplibre GL JavaScript con funzionalità di geocodifica di Amazon Location. Utilizza l'autenticazione tramite Amazon Cognito Identity Pool per accedere alle risorse di Amazon Location. La mappa viene renderizzata con uno stile e delle coordinate centrali specificati e consente di cercare luoghi sulla mappa.

```
// Import MapLibre GL JS
import maplibregl from "maplibre-gl";
// Import from the AWS JavaScript SDK V3
import { LocationClient } from "@aws-sdk/client-location";
// Import the utility functions
import { withIdentityPoolId } from "@aws/amazon-location-utilities-auth-helper";
// Import the AmazonLocationWithMaplibreGeocoder
import { buildAmazonLocationMaplibreGeocoder, AmazonLocationMaplibreGeocoder } from
"@aws/amazon-location-for-maplibre-gl-geocoder"

const identityPoolId = "Identity Pool ID";
const mapName = "Map Name";
const region = "Region"; // region containing the Amazon Location resource
const placeIndex = "PlaceIndexName" // Name of your places resource in your AWS
Account.

// Create an authentication helper instance using credentials from Amazon Cognito
const authHelper = await withIdentityPoolId("Identity Pool ID");

const client = new LocationClient({
  region: "Region", // Region containing Amazon Location resources
  ...authHelper.getLocationClientConfig(), // Configures the client to use
  credentials obtained via Amazon Cognito
```

```
});

// Render the map
const map = new maplibregl.Map({
  container: "map",
  center: [-123.115898, 49.295868],
  zoom: 10,
  style: `https://maps.geo.${region}.amazonaws.com/maps/v0/maps/${mapName}/style-
descriptor`,
  ...authHelper.getMapAuthenticationOptions(),
});

// Gets an instance of the AmazonLocationMaplibreGeocoder Object.
const amazonLocationMaplibreGeocoder = buildAmazonLocationMaplibreGeocoder(client,
placeIndex, {enableAll: true});

// Now we can add the Geocoder to the map.
map.addControl(amazonLocationMaplibreGeocoder.getPlacesGeocoder());
```

## Utilizzo con un browser

Questo esempio utilizza Amazon Location Client per effettuare una richiesta di autenticazione tramite Amazon Cognito.

### Note

Alcuni di questi esempi utilizzano Amazon Location Client. Amazon Location Client si basa sull'[AWS SDK per JavaScript V3](#) e consente di effettuare chiamate ad Amazon Location tramite uno script a cui si fa riferimento in un file HTML.

Includi quanto segue in un file HTML:

```
< Import the Amazon Location With Maplibre Geocoder>
<script src="https://www.unpkg.com/@aws/amazon-location-with-maplibre-geocoder@1"></
script>
<Import the Amazon Location Client>
<script src="https://www.unpkg.com/@aws/amazon-location-client@1"></script>
<!Import the utility library>
<script src="https://www.unpkg.com/@aws/amazon-location-utilities-auth-helper@1"></
script>
```

Includi quanto segue in un JavaScript file:

```
const identityPoolId = "Identity Pool ID";
const mapName = "Map Name";
const region = "Region"; // region containing Amazon Location resource

// Create an authentication helper instance using credentials from Amazon Cognito
const authHelper = await
  amazonLocationAuthHelper.withIdentityPoolId(identityPoolId);

// Render the map
const map = new maplibregl.Map({
  container: "map",
  center: [-123.115898, 49.295868],
  zoom: 10,
  style: `https://maps.geo.${region}.amazonaws.com/maps/v0/maps/${mapName}/style-
descriptor`,
  ...authHelper.getMapAuthenticationOptions(),
});

// Initialize the AmazonLocationMaplibreGeocoder object
const amazonLocationMaplibreGeocoderObject =
  amazonLocationMaplibreGeocoder.buildAmazonLocationMaplibreGeocoder(client,
  placesName, {enableAll: true});

// Use the AmazonLocationWithMaplibreGeocoder object to add a geocoder to the map.
map.addControl(amazonLocationMaplibreGeocoderObject.getPlacesGeocoder());
```

Di seguito sono elencate le funzioni e i comandi utilizzati nel plug-in di MapLibre geocodifica Amazon Location:

- **buildAmazonLocationMaplibreGeocoder**

Questa classe crea un'istanza di `AmazonLocationMaplibreGeocoder` che è il punto di ingresso per tutte le altre chiamate:

```
const amazonLocationMaplibreGeocoder = buildAmazonLocationMaplibreGeocoder(client,
  placesIndex, {enableAll: true});
```

- **getPlacesGeocoder**



Restituisce un oggetto iControl pronto all'uso che può essere aggiunto direttamente a una mappa.

```
const geocoder = getPlacesGeocoder();

// Initialize map
let map = await initializeMap();

// Add the geocoder to the map.
map.addControl(geocoder);
```

## Amazon Location SDK e librerie

L'Amazon Location SDK è un insieme di librerie open source che forniscono funzionalità utili per lo sviluppo di applicazioni Amazon Location. Sono incluse le seguenti funzionalità:

- Client Amazon Location: gli oggetti Amazon Location nell' AWS SDK v3 sono raggruppati e impacchettati per facilitare l'uso nello sviluppo web.
- Autenticazione: l'utilità di autenticazione semplifica l'autenticazione (utilizzando Amazon Cognito o chiavi API) durante la creazione di una pagina [JavaScriptWeb](#) o un'[applicazione](#) iOS o Android per Amazon Location Service.
- Tracciamento: gli SDK di tracciamento mobile sono disponibili per [iOS](#) e [Android](#). Questo SDK semplifica l'interazione delle applicazioni mobili con Amazon Location Trackers.
- [Funzioni GeoJSON di Amazon Location: le utilità di conversione GeoJSON semplificano la conversione tra i dati formattati GeoJSON standard del settore e i formati API di Amazon Location.](#)

### Argomenti

- [Come iniziare a utilizzare l'Amazon Location SDK](#)
- [Client di localizzazione Amazon](#)
- [JavaScript Aiutante di autenticazione](#)
- [Aiutanti di conversione GeoJSON](#)
- [SDK di autenticazione per dispositivi mobili Android](#)
- [SDK di autenticazione per dispositivi mobili iOS](#)
- [SDK di monitoraggio per dispositivi mobili Android](#)
- [SDK di monitoraggio per dispositivi mobili iOS](#)

## Come iniziare a utilizzare l'Amazon Location SDK

L'Amazon Location SDK è un insieme di funzioni che possono semplificare l'utilizzo di Amazon Location Service in un'applicazione. Puoi installare e importare queste funzioni nella tua JavaScript applicazione. Le seguenti sezioni descrivono il client Amazon Location e le librerie di supporto GeoJSON e di autenticazione.

### Client di localizzazione Amazon

Con AWS SDK v3, l'SDK è separato per servizio. Puoi installare solo le parti di cui hai bisogno. Ad esempio, per installare il client Amazon Location e il provider di credenziali per Amazon Cognito, utilizza i seguenti comandi.

```
npm install @aws-sdk/client-location
npm install @aws-sdk/credential-providers
```

Per facilitare l'utilizzo di Amazon Location Service nelle applicazioni JavaScript web frontend, AWS fornisce un pacchetto ospitato della libreria Amazon Location e del fornitore di credenziali. Per utilizzare il client in bundle, aggiungilo al codice HTML in un tag script, come segue:

```
<script src="https://unpkg.com/@aws/amazon-location-client@1.x/dist/amazonLocationClient.js"></script>
```

#### Note

Il pacchetto è aggiornato e retrocompatibile per facilitarne l'uso. Utilizzando questo tag di script o l'installazione di NPM otterrai sempre la versione più recente.

### JavaScript Aiutante di autenticazione

L'helper di JavaScript autenticazione Amazon Location semplifica l'autenticazione quando effettui chiamate API Amazon Location dalla tua applicazione. JavaScript Questo assistente di autenticazione ti aiuta specificamente a utilizzare [Amazon Cognito](#) o [le chiavi API](#) come metodo di autenticazione. Questa è una libreria open source disponibile qui: [https://github.com/aws-geospatial/GitHub amazon-location-utilities-auth -helper-js](https://github.com/aws-geospatial/GitHub-amazon-location-utilities-auth-helper-js).

### Note

Il supporto di Amazon Cognito nell'helper di autenticazione non supporta la funzionalità di identità federate di Amazon Cognito.

## Installazione

Puoi usare le librerie con un'installazione locale, se usi un sistema di compilazione come webpack, o includendo JavaScript pacchetti predefiniti con `<script>` tag nel tuo html.

- Usa il seguente comando per installare la libreria, usando NPM:

```
npm install @aws/amazon-location-utilities-auth-helper
```

- Utilizzate il seguente comando nel file HTML per caricare lo script:

```
<script src="https://unpkg.com/@aws/amazon-location-utilities-auth-helper@1.x/dist/amazonLocationAuthHelper.js"></script>
```

## Importa

Per utilizzare una funzione specifica nell' JavaScript applicazione, è necessario importare tale funzione. Il codice seguente viene utilizzato per importare la funzione `withIdentityPoolId` nell'applicazione.

```
import { withIdentityPoolId } from '@aws/amazon-location-utilities-auth-helper';
```

## Funzioni di autenticazione

Gli helper di autenticazione di Amazon Location includono le seguenti funzioni che restituiscono un `AuthHelper` oggetto:

- `async withIdentityPoolId( identityPoolId: string): AuthHelper`— Questa funzione restituisce un `AuthHelper` oggetto, inizializzato per funzionare con Amazon Cognito
- `async withAPIKey( API_KEY: string): AuthHelper`— Questa funzione restituisce un `AuthHelper` oggetto, inizializzato per funzionare con le chiavi API.

L'`AuthHelper` oggetto fornisce le seguenti funzioni:

- `AuthHelper.getMapAuthenticationOptions()`— Questa funzione dell'`AuthHelper` oggetto restituisce un JavaScript oggetto con le opzioni della mappa `transformRequest` che può essere utilizzato con le opzioni della mappa in MapLibre JS. Fornito solo se inizializzato con un pool di identità.
- `AuthHelper.getLocationClientConfig()`— Questa funzione dell'`AuthHelper` oggetto restituisce un JavaScript oggetto con il `credentials` quale è possibile inizializzare un `LocationClient`
- `AuthHelper.getCredentials()`— Questa funzione dell'`AuthHelper` oggetto restituisce le credenziali interne di Amazon Cognito. Fornito solo se inizializzato con un pool di identità.

Esempio: inizializzazione di un oggetto MapLibre mappa con Amazon Cognito, utilizzando un `AuthHelper`

```
import { withIdentityPoolId } from '@aws/amazon-location-utilities-auth-helper';

const authHelper = await withIdentityPoolId("identity-pool-id"); // use Cognito pool id
for credentials

const map = new maplibregl.Map({
  container: "map", // HTML element ID of map element
  center: [-123.1187, 49.2819], // initial map center point
  zoom: 16, // initial map zoom
  style: https://maps.geo.region.amazonaws.com/maps/v0/maps/mapName/style-
descriptor', // Defines the appearance of the map
  ...authHelper.getMapAuthenticationOptions(), // Provides credential options
  required for requests to Amazon Location
});
```

Esempio: inizializzazione dell'oggetto della MapLibre mappa con una chiave API (in questo caso non `AuthHelper` è necessaria)

```
const map = new maplibregl.Map({
  container: "map", // HTML element ID of map element
  center: [-123.1187, 49.2819], // initial map center point
  zoom: 16, // initial map zoom
  style: https://maps.geo.region.amazonaws.com/maps/v0/maps/${mapName}/style-
descriptor?key=api-key-id',
```

```
});
```

Esempio: inizializza il client Location dall' AWS SDK per JS, utilizzando Amazon Cognito e AuthHelper

Questo esempio utilizza AWS SDK per la versione 3. JavaScript

```
import { withIdentityPoolId } from '@aws/amazon-location-utilities-auth-helper';

const authHelper = await withIdentityPoolId("identity-pool-id"); // use Cognito pool id
for credentials

//initialize the Location client:
const client = new LocationClient({
  region: "region",
  ...authHelper.getLocationClientConfig() // sets up the Location client to use the
  Cognito pool defined above
});

//call a search function with the location client:
const result = await client.send(new SearchPlaceIndexForPositionCommand({
  IndexName: "place-index", // Place index resource to use
  Position: [-123.1187, 49.2819], // position to search near
  MaxResults: 10 // number of results to return
}));
```

Esempio: inizializza il client Location dall' AWS SDK per JS, utilizzando una chiave API e AuthHelper

Questo esempio utilizza AWS SDK per la versione 3. JavaScript

```
import { withAPIKey } from '@aws/amazon-location-utilities-auth-helper';

const authHelper = await withAPIKey("api-key-id"); // use API Key id for credentials

//initialize the Location client:
const client = new LocationClient({
  region: "region",
  ...authHelper.getLocationClientConfig() // sets up the Location client to use the
  API Key defined above
});

//call a search function with the location client:
```

```
const result = await client.send(new SearchPlaceIndexForPositionCommand({
  IndexName: "place-index", // Place index resource to use
  Position: [-123.1187, 49.2819], // position to search near
  MaxResults: 10 // number of results to return
}));
```

## Aiutanti di conversione GeoJSON

[Gli helper di conversione Amazon Location GeoJSON](#) forniscono strumenti per convertire i tipi di dati di Amazon Location Service da e verso il formato GeoJSON standard del settore. GeoJSON viene utilizzato, ad esempio, MapLibre con per visualizzare i dati geografici sulla mappa. Questa è una libreria open source disponibile qui: <https://github.com/aws-geospatial/amazon-location-utilities-datatypes> -js. GitHub

### Installazione

Puoi usare le librerie con un'installazione locale, come webpack, o includendo JavaScript pacchetti predefiniti con `<script>` tag nel tuo html.

- Usa il seguente comando per installare la libreria, usando NPM.

```
npm install @aws/amazon-location-utilities-datatypes
```

- Utilizzate il seguente comando nel file HTML per caricare lo script:

```
<script src="https://unpkg.com/@aws/amazon-location-utilities-datatypes@1.x/dist/amazonLocationDataConverter.js"></script>
```

### Importa

Per utilizzare una funzione specifica nell' JavaScript applicazione, è necessario importare tale funzione. Il codice seguente viene utilizzato per importare la funzione `placeToFeatureCollection` nell'applicazione.

```
import { placeToFeatureCollection } from '@aws/amazon-location-utilities-datatypes';
```

## Funzioni di conversione GeoJSON

Gli helper di conversione GeoJSON di Amazon Location includono le seguenti funzioni:

- `placeToFeatureCollection(place: GetPlaceResponse | searchPlaceIndexForPositionResponse | searchPlaceIndexForTextResponse, keepNull: boolean): FeatureCollection`— Questa funzione converte le risposte dalle funzioni di ricerca dei luoghi in un `FeatureCollection` GeoJSON con 1 o più funzioni `Point`.
- `devicePositionToFeatureCollection(devicePositions: GetDevicePositionResponse | BatchGetDevicePositionResponse | GetDevicePositionHistoryResponse | ListDevicePositionsResponse, keepNull: boolean): FeatureCollection`— Questa funzione converte le risposte dalle funzioni di posizione del dispositivo tracker in un `GeoJSON FeatureCollection` con 1 o più funzioni `Point`.
- `routeToFeatureCollection(legs: CalculateRouteResponse): FeatureCollection`— Questa funzione converte le risposte dalla funzione di calcolo del percorso in un `FeatureCollection` GeoJSON con una singola funzionalità. `MultiStringLine` Ogni tappa del percorso è rappresentata da una `LineString` voce in. `MultiStringLine`
- `geofenceToFeatureCollection(geofences: GetGeofenceResponse | PutGeofenceRequest | BatchPutGeofenceRequest | ListGeofencesResponse): FeatureCollection`— Questa funzione converte la richiesta o la risposta delle funzioni di geofence in un `GeoJSON` con funzionalità `Polygon`. `FeatureCollection` Può convertire i geofence sia nella risposta che nella richiesta, consentendoti di mostrare i geofence su una mappa prima di caricarli con `or. PutGeofence BatchPutGeofence`

Questa funzione convertirà un geofence circolare in un elemento con un poligono approssimato, ma avrà anche le proprietà «centro» e «raggio» per ricreare il geofence circolare, se necessario (vedere la funzione successiva).

- `featureCollectionToGeofences(featureCollection: FeatureCollection): BatchPutGeofenceRequestEntry[]`— Questa funzione converte un `FeatureCollection` GeoJSON con funzionalità `Polygon` in una matrice `BatchPutGeofenceRequestEntry` di oggetti, in modo che il risultato possa essere utilizzato per creare una richiesta per. `BatchPutGeofence`

Se una `Feature` in `FeatureCollection` ha le proprietà «centro» e «raggio», verrà convertita in una voce di richiesta di geofence circolare, ignorando la geometria del poligono.

Esempio: converti i risultati della ricerca in un livello di punti in MapLibre

Questo esempio utilizza AWS SDK per la versione 3 JavaScript .

```
import { placeToFeatureCollection } from '@aws/amazon-location-utility-datatypes';
```

```
...

let map; // map here is an initialized MapLibre instance

const client = new LocationClient(config);
const input = { your_input };
const command = new searchPlaceIndexForTextCommand(input);
const response = await client.send(command);

// calling utility function to convert the response to GeoJSON
const featureCollection = placeToFeatureCollection(response);
map.addSource("search-result", featureCollection);
map.addLayer({
  id: "search-result",
  type: "circle",
  source: "search-result",
  paint: {
    "circle-radius": 6,
    "circle-color": "#B42222",
  },
});
```

## SDK di autenticazione per dispositivi mobili Android

Queste utilità ti aiutano ad autenticarti quando effettui chiamate API Amazon Location Service dalle tue applicazioni Android. Ciò è particolarmente utile quando si utilizzano [Amazon Cognito](#) o [le chiavi API](#) come metodo di autenticazione.

L'SDK di autenticazione mobile Android è disponibile su github: Amazon [Location Service Mobile Authentication SDK per Android](#). [Inoltre, sia l'SDK di autenticazione mobile che l' AWS SDK sono disponibili nel repository Maven.AWS](#)

### Installazione

Per utilizzare l'SDK di autenticazione mobile, aggiungi le seguenti istruzioni di importazione al tuo file in Android Studio. `build.gradle`

```
implementation("software.amazon.location:auth:0.0.1")
implementation("com.amazonaws:aws-android-sdk-location:2.72.0")
```

### Funzioni di autenticazione

L'Authentication Helper SDK ha le seguenti funzioni:



- `authHelper.authenticateWithApiKey("My-Amazon-Location-API-Key")`: `LocationCredentialsProvider`: Questa funzione restituisce una chiave API `LocationCredentialsProvider` inizializzata per funzionare con una chiave API.
- `authHelper.authenticateWithCognitoIdentityPool("My-Cognito-Identity-Pool-Id")`: `LocationCredentialsProvider`: Questa funzione restituisce un file `LocationCredentialsProvider` inizializzato per funzionare con un pool di identità di Amazon Cognito.

## Utilizzo

Per utilizzare l'SDK nel codice, importa le seguenti classi:

```
import com.amazonaws.services.geo.AmazonLocationClient
import software.amazon.location.auth.AuthHelper
import software.amazon.location.auth.LocationCredentialsProvider
```

Hai due opzioni per creare l'helper di autenticazione e le istanze del provider di client di localizzazione. Puoi creare un'istanza utilizzando le [chiavi API di Amazon Location](#) o [Amazon Cognito](#).

- Per creare un'istanza helper di autenticazione utilizzando una chiave API Amazon Location, dichiara la classe helper come segue:

```
var authHelper = AuthHelper(applicationContext)
var locationCredentialsProvider : LocationCredentialsProvider =
    authHelper.authenticateWithApiKey("My-Amazon-Location-API-Key")
```

- Per creare un'istanza helper di autenticazione utilizzando Amazon Cognito, dichiara la classe helper come segue:

```
var authHelper = AuthHelper(applicationContext)
var locationCredentialsProvider : LocationCredentialsProvider =
    authHelper.authenticateWithCognitoIdentityPool("My-Cognito-Identity-Pool-Id")
```

Puoi creare un'istanza client Amazon Location utilizzando il provider di credenziali di posizione ed effettuare chiamate al servizio Amazon Location. L'esempio seguente cerca luoghi vicini a una latitudine e una longitudine specificate.

```
var locationClient =
    authHelper.getLocationClient(locationCredentialsProvider.getCredentialsProvider())
var searchPlaceIndexForPositionRequest =
    SearchPlaceIndexForPositionRequest().withIndexName("My-Place-Index-
    Name").withPosition(arrayListOf(30.405423, -97.718833))
var nearbyPlaces =
    locationClient.searchPlaceIndexForPosition(searchPlaceIndexForPositionRequest)
```

## SDK di autenticazione per dispositivi mobili iOS

Queste utilità ti aiutano ad autenticarti quando effettui chiamate API Amazon Location Service dalle tue applicazioni iOS. Ciò è particolarmente utile quando si utilizzano [Amazon Cognito](#) o [le chiavi API](#) come metodo di autenticazione.

L'SDK di autenticazione mobile iOS è disponibile su github: Amazon [Location Service Mobile Authentication SDK per iOS](#).

### Installazione

Installa l'SDK in un progetto Xcode:

1. Vai su File, quindi seleziona Add Package Dependencies nel tuo progetto XCode.
2. Digita l'URL del pacchetto: <https://github.com/aws-geospatial/amazon-location-mobile-auth-sdk-ios/> nella barra di ricerca e premi il tasto invio.
3. Seleziona il amazon-location-mobile-auth-sdk-ios pacchetto e premi Aggiungi pacchetto.
4. Seleziona il prodotto del AmazonLocationiOSAuthSDK pacchetto e premi Aggiungi pacchetto.

### Funzioni di autenticazione

L'Authentication Helper SDK ha le seguenti funzioni:

- `authHelper.authenticateWithApiKey("My-Amazon-Location-API-Key")`: `LocationCredentialsProvider`: Questa funzione restituisce una chiave API `LocationCredentialsProvider` inizializzata per funzionare con una chiave API.
- `authHelper.authenticateWithCognitoIdentityPool("My-Cognito-Identity-Pool-Id")`: `LocationCredentialsProvider`: Questa funzione restituisce un file `LocationCredentialsProvider` inizializzato per funzionare con un pool di identità di Amazon Cognito.

## Utilizzo

Per utilizzare l'SDK di autenticazione mobile, aggiungi le seguenti istruzioni alla tua attività:

```
import AmazonLocationiOSAuthSDK
import AWSLocationXCF
```

Hai due opzioni per creare l'helper di autenticazione e le istanze del provider di client di localizzazione. Puoi creare un'istanza utilizzando le [chiavi API di Amazon Location](#) o [Amazon Cognito](#).

- Per creare un'istanza helper di autenticazione utilizzando una chiave API Amazon Location, dichiara la classe helper come segue:

```
let authHelper = AuthHelper()
let locationCredentialsProvider = authHelper.authenticateWithAPIKey(apiKey: "My-Amazon-Location-API-Key", region: "account-region")
```

- Per creare un'istanza helper di autenticazione utilizzando Amazon Cognito, dichiara la classe helper come segue:

```
let authHelper = AuthHelper()
let locationCredentialsProvider =
  authHelper.authenticateWithCognitoUserPool(identityPoolId: "My-Amazon-Location-API-Key", region: "account-region")
```

Puoi creare un'istanza client Amazon Location utilizzando il provider di credenziali di posizione ed effettuare chiamate al servizio Amazon Location. L'esempio seguente cerca luoghi vicini a una latitudine e una longitudine specificate.

```
let locationClient = AWSLocation.default()
let searchPlaceIndexForPositionRequest =
  AWSLocationSearchPlaceIndexForPositionRequest()!
searchPlaceIndexForPositionRequest.indexName = "My-Place-Index-Name"
searchPlaceIndexForPositionRequest.position = [30.405423, -97.718833]
let nearbyPlaces = locationClient.searchPlaceIndex(forPosition:
  searchPlaceIndexForPositionRequest)
```

## SDK di monitoraggio per dispositivi mobili Android

L'SDK di tracciamento mobile Amazon Location fornisce utilità che aiutano a autenticare, acquisire le posizioni dei dispositivi e inviare aggiornamenti di posizione ad Amazon Location Tracker con facilità. L'SDK supporta il filtraggio locale degli aggiornamenti di posizione con intervalli di aggiornamento configurabili. Ciò riduce i costi dei dati e ottimizza la connettività intermittente per le applicazioni Android.

L'SDK di tracciamento Android è disponibile su GitHub: [Amazon Location Mobile Tracking SDK](#) per Android. [Inoltre, sia l'SDK di autenticazione mobile che l' AWS SDK sono disponibili nel repository Maven.AWS](#) L'SDK di tracciamento Android è progettato per funzionare con l'SDK generale. AWS

Questa sezione tratta i seguenti argomenti per l'SDK Android per il tracciamento mobile di Amazon Location:

### Argomenti

- [Installazione](#)
- [Utilizzo](#)
- [Filtri](#)
- [Funzioni di tracciamento Android Mobile SDK](#)
- [Esempi](#)

### Installazione

Per installare l'SDK, aggiungi le seguenti righe alla sezione delle dipendenze del tuo file build.gradle in Android Studio:

```
implementation("software.amazon.location:tracking:0.0.1")
implementation("software.amazon.location:auth:0.0.1")
implementation("com.amazonaws:aws-android-sdk-location:2.72.0")
```

### Utilizzo

Questa procedura mostra come utilizzare l'SDK per autenticare e creare l'oggetto:

LocationTracker

**Note**

Questa procedura presuppone che tu abbia importato la libreria menzionata nella sezione.

[Installazione](#)

1. Importa le seguenti classi nel tuo codice:

```
import software.amazon.location.tracking.LocationTracker
import software.amazon.location.tracking.config.LocationTrackerConfig
import software.amazon.location.tracking.util.TrackingSdkLogLevel
import com.amazonaws.services.geo.AmazonLocationClient
import software.amazon.location.auth.AuthHelper
import software.amazon.location.auth.LocationCredentialsProvider
```

2. Quindi crea un `AuthHelper`, poiché il `LocationCredentialsProvider` parametro è necessario per creare un `LocationTracker` oggetto:

```
// Create an authentication helper using credentials from Cognito
val authHelper = AuthHelper(applicationContext)
val locationCredentialsProvider : LocationCredentialsProvider =
    authHelper.authenticateWithCognitoIdentityPool("My-Cognito-Identity-Pool-Id")
```

3. Ora, usate il `LocationCredentialsProvider` comando and `LocationTrackerConfig` per creare un `LocationTracker` oggetto:

```
val config = LocationTrackerConfig(
    trackerName = "MY-TRACKER-NAME",
    logLevel = TrackingSdkLogLevel.DEBUG,
    accuracy = Priority.PRIORITY_HIGH_ACCURACY,
    latency = 1000,
    frequency = 5000,
    waitForAccurateLocation = false,
    minUpdateIntervalMillis = 5000,
)
locationTracker = LocationTracker(
    applicationContext,
    locationCredentialsProvider,
    config,
)
```

## Filtri

L'SDK Android per il tracciamento mobile di Amazon Location dispone di tre filtri di posizione integrati.

- `TimeLocationFilter`: filtra la posizione corrente da caricare in base a un intervallo di tempo definito.
- `DistanceLocationFilter`: filtra gli aggiornamenti della posizione in base a una soglia di distanza specificata.
- `AccuracyLocationFilter`: filtra gli aggiornamenti della posizione confrontando la distanza percorsa dall'ultimo aggiornamento con la precisione della posizione corrente.

Questo esempio aggiunge filtri in `LocationTracker` al momento della creazione:

```
val config = LocationTrackerConfig(
    trackerName = "MY-TRACKER-NAME",
    logLevel = TrackingSdkLogLevel.DEBUG,
    accuracy = Priority.PRIORITY_HIGH_ACCURACY,
    latency = 1000,
    frequency = 5000,
    waitForAccurateLocation = false,
    minUpdateIntervalMillis = 5000,
    locationFilters = mutableListOf(TimeLocationFilter(), DistanceLocationFilter(),
    AccuracyLocationFilter())
)
locationTracker = LocationTracker(
    applicationContext,
    locationCredentialsProvider,
    config,
)
```

Questo esempio abilita e disabilita il filtro in fase di esecuzione con `LocationTracker`:

```
// To enable the filter
locationTracker?.enableFilter(TimeLocationFilter())

// To disable the filter
locationTracker?.disableFilter(TimeLocationFilter())
```

## Funzioni di tracciamento Android Mobile SDK

L'SDK di tracciamento mobile Amazon Location per Android include le seguenti funzioni:

- Classe: `LocationTracker`

```
constructor(context: Context, locationCredentialsProvider:
LocationCredentialsProvider, trackerName: String), oppure
constructor(context: Context, locationCredentialsProvider:
LocationCredentialsProvider, clientConfig: LocationTrackerConfig)
```

Questa è una funzione di inizializzazione per creare un `LocationTracker` oggetto. Richiede istanze di `trackerName` e `LocationCredentialsProvider`, facoltativamente, un'istanza di `LocationTrackingConfig`. Se la configurazione non viene fornita, verrà inizializzata con valori predefiniti.

- Classe: `LocationTracker`

```
start(locationTrackingCallback: LocationTrackingCallback)
```

Avvia il processo di accesso alla posizione dell'utente e di invio della stessa a un localizzatore di localizzazione Amazon.

- Classe: `LocationTracker`

```
isTrackingInForeground()
```

Verifica se il tracciamento della posizione è attualmente in corso.

- Classe: `LocationTracker`

```
stop()
```

Interrompe il processo di tracciamento della posizione dell'utente.

- Classe: `LocationTracker`

```
startTracking()
```

Avvia il processo di accesso alla posizione dell'utente e di invio al AWS tracker.

- Classe: `LocationTracker`

```
startBackground(mode: BackgroundTrackingMode, serviceCallback:
ServiceCallback)
```

Avvia il processo di accesso alla posizione dell'utente e di invio al AWS tracker mentre l'applicazione è in background. BackgroundTrackingMode ha le seguenti opzioni:

- ACTIVE\_TRACKING: questa opzione tiene traccia attivamente degli aggiornamenti sulla posizione di un utente.
- BATTERY\_SAVER\_TRACKING: Questa opzione tiene traccia degli aggiornamenti sulla posizione dell'utente ogni 15 minuti.
- Classe: LocationTracker

```
stopBackgroundService()
```

Interrompe il processo di accesso alla posizione dell'utente e di invio della stessa al AWS tracker mentre l'applicazione è in background.

- Classe: LocationTracker

```
getTrackerDeviceLocation()
```

Recupera la posizione del dispositivo dai servizi di localizzazione di Amazon.

- Classe: LocationTracker

```
getDeviceLocation(locationTrackingCallback: LocationTrackingCallback?)
```

Recupera la posizione corrente del dispositivo dal client del fornitore di geolocalizzazione fusa e la carica su Amazon Location tracker.

- Classe: LocationTracker

```
uploadLocationUpdates(locationTrackingCallback:  
LocationTrackingCallback?)
```

Carica la posizione del dispositivo nei servizi di localizzazione Amazon dopo averla filtrata in base ai filtri di posizione configurati.

- Classe: LocationTracker

```
enableFilter(filter: LocationFilter)
```

Abilita un particolare filtro di posizione.

- Classe: LocationTracker

```
checkFilterIsExistsAndUpdateValue(filter: LocationFilter)
```



Disabilita un filtro di posizione particolare.

- Classe: `LocationTrackerConfig`

```
LocationTrackerConfig( // Required var trackerName:
String, // Optional var locationFilters: MutableList =
mutableListOf( TimeLocationFilter(), DistanceLocationFilter(), ), var
logLevel: TrackingSdkLogLevel = TrackingSdkLogLevel.DEBUG, var accuracy:
Int = Priority.PRIORITY_HIGH_ACCURACY, var latency: Long = 1000, var
frequency: Long = 1500, var waitForAccurateLocation: Boolean = false, var
minUpdateIntervalMillis: Long = 1000, var persistentNotificationConfig:
NotificationConfig = NotificationConfig())
```

Questo inizializza il `LocationTrackerConfig` con valori dei parametri definiti dall'utente. Se non viene fornito un valore di parametro, verrà impostato su un valore predefinito.

- Classe: `LocationFilter`

```
shouldUpload(currentLocation: LocationEntry, previousLocation:
LocationEntry?): Boolean
```

`LocationFilter` È un protocollo che gli utenti possono implementare per l'implementazione di filtri personalizzati. È necessario implementare la `shouldUpload` funzione per confrontare la posizione precedente e quella corrente e restituire se la posizione corrente deve essere caricata.

## Esempi

Il seguente esempio di codice mostra la funzionalità dell'SDK di tracciamento mobile.

Questo esempio utilizza `LocationTracker` per avviare e interrompere il tracciamento in background:

```
// For starting the location tracking
locationTracker?.startBackground(
BackgroundTrackingMode.ACTIVE_TRACKING,
object : ServiceCallback {
    override fun serviceStopped() {
        if (selectedTrackingMode == BackgroundTrackingMode.ACTIVE_TRACKING) {
            isLocationTrackingBackgroundActive = false
        } else {
            isLocationTrackingBatteryOptimizeActive = false
        }
    }
})
```

```
    }  
  }  
},  
)  
  
// For stopping the location tracking  
locationTracker?.stopBackgroundService()
```

## SDK di monitoraggio per dispositivi mobili iOS

L'SDK di tracciamento mobile Amazon Location fornisce utilità che aiutano a autenticare, acquisire le posizioni dei dispositivi e inviare aggiornamenti di posizione ad Amazon Location Tracker con facilità. L'SDK supporta il filtraggio locale degli aggiornamenti di posizione con intervalli di aggiornamento configurabili. Ciò riduce i costi dei dati e ottimizza la connettività intermittente per le applicazioni iOS.

L'SDK di tracciamento iOS è disponibile su GitHub: [Amazon Location Mobile Tracking SDK](#) per iOS.

Questa sezione tratta i seguenti argomenti per l'SDK iOS per il tracciamento mobile di Amazon Location:

### Argomenti

- [Installazione](#)
- [Utilizzo](#)
- [Filtri](#)
- [Funzioni di tracciamento dell'SDK per dispositivi mobili iOS](#)
- [Esempi](#)

### Installazione

Utilizza la seguente procedura per installare l'SDK di tracciamento mobile per iOS:

1. Nel tuo progetto Xcode, vai su File e seleziona Add Package Dependencies.
2. Digita il seguente URL: <https://github.com/aws-geospatial/amazon-location-mobile-tracking-sdk-ios/> nella barra di ricerca e premi il tasto invio.
3. Seleziona il amazon-location-mobile-tracking-sdk-ios pacchetto e fai clic su Aggiungi pacchetto.
4. Seleziona il prodotto del AmazonLocationiOSTrackingSDK pacchetto e fai clic su Aggiungi pacchetto.

## Utilizzo

La procedura seguente mostra come creare un helper di autenticazione utilizzando le credenziali di Cognito.

1. Dopo aver installato la libreria, devi aggiungere una o entrambe le descrizioni nel file: `info.plist`

```
Privacy - Location When In Use Usage Description
Privacy - Location Always and When In Use Usage Description
```

2. Quindi, importali `AuthHelper` nella tua classe:

```
import AmazonLocationiOSAuthSDKimport AmazonLocationIOSTrackingSDK
```

3. Quindi creerai un `AuthHelper` oggetto e lo utilizzerai con l' AWS SDK, creando un helper di autenticazione utilizzando le credenziali di Amazon Cognito.

```
let authHelper = AuthHelper()
let locationCredentialsProvider =
  authHelper.authenticateWithCognitoUserPool(identityPoolId: "My-Cognito-Identity-
  Pool-Id", region: "My-region") //example: us-east-1
let locationTracker = LocationTracker(provider: locationCredentialsProvider,
  trackerName: "My-tracker-name")

// Optionally you can set ClientConfig with your own values in either initialize or
  in a separate function
// let trackerConfig = LocationTrackerConfig(locationFilters:
  [TimeLocationFilter(), DistanceLocationFilter()],
  trackingDistanceInterval: 30,
  trackingTimeInterval: 30,
  logLevel: .debug)

// locationTracker = LocationTracker(provider: credentialsProvider, trackerName:
  "My-tracker-name",config: trackerConfig)
// locationTracker.setConfig(config: trackerConfig)
```

## Filtri

L'SDK iOS per il tracciamento mobile di Amazon Location dispone di tre filtri di posizione integrati.

- **TimeLocationFilter**: filtra la posizione corrente da caricare in base a un intervallo di tempo definito.
- **DistanceLocationFilter**: filtra gli aggiornamenti della posizione in base a una soglia di distanza specificata.
- **AccuracyLocationFilter**: filtra gli aggiornamenti della posizione confrontando la distanza percorsa dall'ultimo aggiornamento con la precisione della posizione corrente.

Questo esempio aggiunge filtri in `LocationTracker` al momento della creazione:

```
val config = LocationTrackerConfig(
    trackerName = "MY-TRACKER-NAME",
    logLevel = TrackingSdkLogLevel.DEBUG,
    accuracy = Priority.PRIORITY_HIGH_ACCURACY,
    latency = 1000,
    frequency = 5000,
    waitForAccurateLocation = false,
    minUpdateIntervalMillis = 5000,
    locationFilters = mutableListOf(TimeLocationFilter(), DistanceLocationFilter(),
    AccuracyLocationFilter())
)

locationTracker = LocationTracker(
    applicationContext,
    locationCredentialsProvider,
    config,
)
```

Questo esempio abilita e disabilita il filtro in fase di esecuzione con `LocationTracker`:

```
// To enable the filter
locationTracker?.enableFilter(TimeLocationFilter())

// To disable the filter
locationTracker?.disableFilter(TimeLocationFilter())
```

## Funzioni di tracciamento dell'SDK per dispositivi mobili iOS

L'SDK di tracciamento mobile Amazon Location per iOS include le seguenti funzioni:

- Classe: `LocationTracker`

```
init(provider: LocationCredentialsProvider, trackerName: String, config: LocationTrackerConfig? = nil)
```

Questa è una funzione di inizializzazione per creare un `LocationTracker` oggetto. Richiede istanze di `trackerName` e `LocationCredentialsProvider`, facoltativamente, un'istanza di `LocationTrackingConfig`. Se la configurazione non viene fornita, verrà inizializzata con valori predefiniti.

- Classe: `LocationTracker`

```
setTrackerConfig(config: LocationTrackerConfig)
```

Questo imposta la configurazione di Tracker in modo che abbia effetto in qualsiasi momento dopo l'inizializzazione del localizzatore di posizione.

- Classe: `LocationTracker`

```
getTrackerConfig()
```

In questo modo viene ottenuta la configurazione di tracciamento della posizione da utilizzare o modificare nell'app.

Restituisce: `LocationTrackerConfig`

- Classe: `LocationTracker`

```
getDeviceId()
```

Ottiene l'ID del dispositivo generato dal localizzatore di posizione.

Restituisce: `String?`

- Classe: `LocationTracker`

```
startTracking()
```

Avvia il processo di accesso alla posizione dell'utente e di invio al AWS tracker.

- Classe: `LocationTracker`

```
resumeTracking()
```

Riprende il processo di accesso alla posizione dell'utente e di invio della stessa al AWS tracker.

- Classe: `LocationTracker`

```
stopTracking()
```

Interrompe il processo di tracciamento della posizione dell'utente.

- Classe: `LocationTracker`

```
startBackgroundTracking(mode: BackgroundTrackingMode)
```

Avvia il processo di accesso alla posizione dell'utente e di invio al AWS tracker mentre l'applicazione è in background. `BackgroundTrackingMode` ha le seguenti opzioni:

- `Active`: Questa opzione non sospende automaticamente gli aggiornamenti della posizione.
- `BatterySaving`: Questa opzione sospende automaticamente gli aggiornamenti della posizione.
- `None`: Questa opzione disabilita nel complesso gli aggiornamenti della posizione in background.

- Classe: `LocationTracker`

```
resumeBackgroundTracking(mode: BackgroundTrackingMode)
```

Riprende il processo di accesso alla posizione dell'utente e di invio al AWS tracker mentre l'applicazione è in background.

- Classe: `LocationTracker`

```
stopBackgroundTracking()
```

Interrompe il processo di accesso alla posizione dell'utente e di invio della stessa al AWS tracker mentre l'applicazione è in background.

- Classe: `LocationTracker`

```
getTrackerDeviceLocation(nextToken: String?, startTime: Date? = nil,  
endTime: Date? = nil, completion: @escaping (Result<GetLocationResponse,  
Error>)
```

Recupera le posizioni di tracciamento caricate per il dispositivo dell'utente tra la data e l'ora di inizio e fine.

Restituisce: `Void`

- Classe: `LocationTrackerConfig`

```
init()
```

Questo inizializza LocationTrackerConfig con valori predefiniti.

- Classe: LocationTrackerConfig

```
init(locationFilters: [LocationFilter]? = nil, trackingDistanceInterval: Double? = nil, trackingTimeInterval: Double? = nil, trackingAccuracyLevel: Double? = nil, uploadFrequency: Double? = nil, desiredAccuracy: CLLocationAccuracy? = nil, activityType: CLActivityType? = nil, logLevel: LogLevel? = nil)
```

Questo inizializza il LocationTrackerConfig con valori dei parametri definiti dall'utente. Se non viene fornito un valore di parametro, verrà impostato su un valore predefinito.

- Classe: LocationFilter

```
shouldUpload(currentLocation: LocationEntity, previousLocation: LocationEntity?, trackerConfig: LocationTrackerConfig)
```

LocationFilter è un protocollo che gli utenti possono implementare per l'implementazione di filtri personalizzati. Un utente dovrebbe implementare una shouldUpload funzione per confrontare la posizione precedente e corrente e restituire se la posizione corrente deve essere caricata.

## Esempi

Questa sezione descrive alcuni esempi di utilizzo dell'Amazon Location Mobile Tracking SDK per iOS.

### Note

Assicurati che le autorizzazioni necessarie siano impostate nel `info.plist` file. Queste sono le stesse autorizzazioni elencate nella [Utilizzo](#) sezione.

L'esempio seguente illustra la funzionalità per tracciare la posizione del dispositivo e recuperare le posizioni tracciate:

```
Privacy - Location When In Use Usage Description  
Privacy - Location Always and When In Use Usage Description
```

**Inizia a tracciare la posizione:**

```
do {
    try locationTracker.startTracking()
    }
catch TrackingLocationError.permissionDenied {
    // Handle permissionDenied by showing the alert message or opening the app
    settings
    }
}
```

**Riprendi a tracciare la posizione:**

```
do {
    try locationTracker.resumeTracking()
    }
catch TrackingLocationError.permissionDenied {
    // Handle permissionDenied by showing the alert message or opening the app settings
    }
}
```

**Smetti di tracciare la posizione:**

```
locationTracker.stopTracking()
```

**Inizia il tracciamento in background:**

```
do {

    locationTracker.startBackgroundTracking(mode: .Active) // .Active, .BatterySaving, .None
    }
catch TrackingLocationError.permissionDenied {
    // Handle permissionDenied by showing the alert message or opening the app settings
    }
}
```

**Riprendi il tracciamento in background:**

```
do {
    locationTracker.resumeBackgroundTracking(mode: .Active)
    }
catch TrackingLocationError.permissionDenied {
    // Handle permissionDenied by showing the alert message or opening the app settings
    }
}
```



```
}
```

Per interrompere il tracciamento in background:

```
locationTracker.stopBackgroundTracking()
```

Recupera le posizioni tracciate del dispositivo dal tracker:

```
func getTrackingPoints(nextToken: String? = nil) {
    let startTime: Date = Date().addingTimeInterval(-86400) // Yesterday's day date and
    time
    let endTime: Date = Date()
    locationTracker.getTrackerDeviceLocation(nextToken: nextToken, startTime: startTime,
    endTime: endTime, completion: { [weak self] result in
        switch result {
            case .success(let response):

                let positions = response.devicePositions
                // You can draw positions on map or use it further as per your requirement

                // If nextToken is available, recursively call to get more data
                if let nextToken = response.nextToken {
                    self?.getTrackingPoints(nextToken: nextToken)
                }
            case .failure(let error):
                print(error)
        }
    })
}
```

## API di localizzazione Amazon

Amazon Location Service fornisce operazioni API per accedere in modo programmatico alla funzionalità di localizzazione. Ciò include le API per mappe, luoghi, percorsi, tracker, geofences e l'etichettatura delle risorse. Per informazioni sulle azioni API disponibili, consulta il [riferimento all'API di Amazon Location Service](#).

Puoi trovare esempi nel [Esempi di codice](#) capitolo di questa guida.

## Utilizzo di Amazon Location con un AWS SDK

AWS I kit di sviluppo software (SDK) sono disponibili per molti linguaggi di programmazione più diffusi. Ogni SDK fornisce un'API, esempi di codice e documentazione che facilita agli sviluppatori la creazione di AWS applicazioni nel linguaggio preferito.

Per ulteriori informazioni sugli SDK disponibili per l'uso con Amazon Location Service per lingua, [SDK per lingua](#) consulta questa guida.

### Versioni SDK

Ti consigliamo di utilizzare la build più recente dell' AWS SDK e qualsiasi altro SDK che utilizzi nei tuoi progetti e di mantenere gli SDK aggiornati. L' AWS SDK offre le caratteristiche e le funzionalità più recenti e anche aggiornamenti di sicurezza. Per trovare la versione più recente dell' AWS SDK per JavaScript, ad esempio, consulta [l'argomento sull'installazione del browser](#) nell'AWS SDK per la documentazione. JavaScript

## Aggiornamenti dei messaggi di errore dell'API Amazon Location

A partire dal 1° agosto 2023, il team di Amazon Location modificherà i messaggi di errore dell'API come descritto nelle tabelle seguenti. I codici di errore non verranno modificati. Se le applicazioni dipendono da stringhe di messaggi di errore esatte, è necessario aggiornare le applicazioni con le nuove stringhe. Per assistenza in caso di domande o problemi, contatta. AWS Support

### Argomenti

- [Luoghi](#)
- [Mappe](#)
- [Localizzatori](#)
- [Route](#)
- [Metadati](#)
- [Geofence](#)

## Luoghi

## Luoghi

Codice di errore	Eccezione	Vecchio messaggio di errore	Nuovo messaggio di errore
500	InternalServerErrorException	Internal Server Exception	Internal server error. Try again later.
404	ResourceNotFoundException	resource <PlaceIndexName> not found, reason: <Reason>  Resource '<PlaceIndexName>' not found  placeIdx<PlaceIndexName> not found, reason: <Reason>  no place index with name '%s' found	Place index not found: <PlaceIndexName>.
404	ResourceNotFoundException	place not found	Place not found: <PlaceId>.
400	ValidationException	PlaceIndex <PlaceIndexName> cannot be used for SearchPlaceIndexForSuggestions because it has IntendedUse <IntendedUse>	A place index with 'IntendedUse' set to Storage does not support 'SearchPlaceIndexForSuggestions' operation.
400	ValidationException	only one of 'BiasPosition' or 'FilterBBox' may be set	Only one of 'BiasPosition' or 'FilterBBox' may be set.

Codice di errore	Eccezione	Vecchio messaggio di errore	Nuovo messaggio di errore
400	ValidationException	BiasPosition must have exactly 2 entries	'BiasPosition' must have exactly 2 entries.
400	ValidationException	BiasPosition[0] must be between -180 and 180	'BiasPosition[0]' must be between -180 and 180.
400	ValidationException	BiasPosition[1] must be between -90 and 90	'BiasPosition[1]' must be between -90 and 90.
400	ValidationException	FilterBBox must have exactly 4 entries	'FilterBBox' must have exactly 4 entries.
400	ValidationException	FilterBBox[0] must be between -180 and 180	'FilterBBox[0]' must be between -180 and 180.
400	ValidationException	FilterBBox[1] must be between -90 and 90	'FilterBBox[1]' must be between -90 and 90.
400	ValidationException	FilterBBox[2] must be between -180 and 180	'FilterBBox[2]' must be between -180 and 180.
400	ValidationException	FilterBBox[3] must be between -90 and 90	'FilterBBox[3]' must be between -90 and 90.
400	ValidationException	FilterBBox must have more southwesterly point before more northeasterly point	'FilterBBox' must have more southwesterly position before more northeasterly position.
400	ValidationException	Position must have exactly 2 entries	'Position' must have exactly 2 entries.

Codice di errore	Eccezione	Vecchio messaggio di errore	Nuovo messaggio di errore
400	ValidationException	Position[0] must be between -180 and 180	'Position[0]' must be between -180 and 180.
400	ValidationException	Position[1] must be between -90 and 90	'Position[1]' must be between -90 and 90.
400	ValidationException	Language is not a valid BCP 47 language tag	'Language' must comply with the BCP 47 Language Tag standard, but was set to <GivenValue>. For more information, see <a href="https://wikipedia.org/wiki/IETF_language_tag">https://wikipedia.org/wiki/IETF_language_tag</a> .
400	ValidationException	'placeID' is invalid	'PlaceId' must be a valid ID.
400	ValidationException	no customer account ID parameter found	'RequesterAccountID' is a required field.
400	ValidationException	Invalid token	'NextToken' must be a valid token.
400	ValidationException	Expired token	'NextToken' must not be expired.
400	ValidationException	unsupported price plan '<PricingPlan>'	'PricingPlan' must be set to RequestBasedUsage.
400	ValidationException	'DataSource' must be one of: Here, Esri	'DataSource' must be one of Esri, Grab, Here.

Codice di errore	Eccezione	Vecchio messaggio di errore	Nuovo messaggio di errore
400	ValidationException	Grab is only supported in the ap-southeast-1 region	'DataSource' Grab must only be used in following regions: ap-southeast-1.
400	ValidationException	'IntendedUse' and 'PricingPlan' must both be provided to update either property	'IntendedUse' and 'PricingPlan' must both be provided to update either attribute .
402	ServiceQuotaExceededException	Place resources per account exceeded quota limits. For more info, see <a href="https://aws.amazon.com/premiumsupport/knowledge-center/manage-service-limits/">https://aws.amazon.com/premiumsupport/knowledge-center/manage-service-limits/</a>	Place index resources have exceeded the quota per account per region. For more information, see <a href="https://aws.amazon.com/premiumsupport/knowledge-center/manage-service-limits/">https://aws.amazon.com/premiumsupport/knowledge-center/manage-service-limits/</a> .
409	ConflictException	Resource already exists	Place index already exists: <PlaceIndexName>.

## Mappe

### Mappe

Codice di errore	Eccezione	Vecchio messaggio di errore	Nuovo messaggio di errore
500	InternalServerErrorException	Internal Server Exception	Internal server error. Try again later.

Codice di errore	Eccezione	Vecchio messaggio di errore	Nuovo messaggio di errore
		unable to find style template  Error fetching style  was not able to serialize the map style file	
404	ResourceNotFoundException	Map not found	Map not found: <MapName>.
404	ResourceNotFoundException	Sprites are not supported for this resource	Sprite not found: <SpriteName>.
400	ValidationException	Resource name should be set	'MapName' is a required field.
400	ValidationException	Must provide a valid number for start and end of Range	Font Unicode range start and end numbers must both be provided.
400	ValidationException	Start of range is an invalid number: <StartValue>	Start of font Unicode range must be a valid number.
400	ValidationException	End of range is an invalid number: <StartValue>	End of font Unicode range must be a valid number.

Codice di errore	Eccezione	Vecchio messaggio di errore	Nuovo messaggio di errore
400	ValidationException	End of range must be exactly 255 higher from start of range, difference found: <Difference>	The difference between the start and end of the font Unicode range must be exactly 255. Difference found: <Difference>.
400	ValidationException	Start of range must be a multiple of 256, found <StartValue>	Start of font Unicode range must be a multiple of 256, but was set to: <StartValue>.
400	ValidationException	Request font is empty	'FontStack' is a required field.
400	ValidationException	Request font is not valid for the datasource <DataSource>	<FontStack> is not a supported font stack for data source <DataSource>. For more information about the list of supported font stacks, see <a href="https://docs.aws.amazon.com/location/latest/APIReference/API_GetMapGlyphs.html">https://docs.aws.amazon.com/location/latest/APIReference/API_GetMapGlyphs.html</a> .



Codice di errore	Eccezione	Vecchio messaggio di errore	Nuovo messaggio di errore
400	ValidationException	Request font is not valid	<FontStack> is not a supported font stack for data source <DataSource>. For more information about the list of supported font stacks, see <a href="https://docs.aws.amazon.com/location/latest/APIReference/API_GetMapGlyphs.html">https://docs.aws.amazon.com/location/latest/APIReference/API_GetMapGlyphs.html</a> .
400	ValidationException	DataSource is invalid: <DataSource>	'DataSource' must be one of Esri, Grab, Here.
400	ValidationException	Request filename is empty	'FileName' is a required field.
400	ValidationException	Request filename is not valid	<SpriteFile> is not a supported sprite file name. For more information about the list of supported sprite file names, see <a href="https://docs.aws.amazon.com/location/latest/APIReference/API_GetMapSprites.html">https://docs.aws.amazon.com/location/latest/APIReference/API_GetMapSprites.html</a> .

Codice di errore	Eccezione	Vecchio messaggio di errore	Nuovo messaggio di errore
400	ValidationException	Filename is invalid: <FileName>	<SpriteFile> is not a supported sprite file name. For more information about the list of supported sprite file names, see <a href="https://docs.aws.amazon.com/location/latest/APIReference/API_GetMapSprites.html">https://docs.aws.amazon.com/location/latest/APIReference/API_GetMapSprites.html</a> .
400	ValidationException	Filename is an invalid content type: <FileName>	<SpriteFile> is not a supported sprite file name. For more information about the list of supported sprite file names, see <a href="https://docs.aws.amazon.com/location/latest/APIReference/API_GetMapSprites.html">https://docs.aws.amazon.com/location/latest/APIReference/API_GetMapSprites.html</a> .
400	ValidationException	Filename is invalid: <FileName>	'Filename' must not be empty.
400	ValidationException	y-coordinate part of 'Y' must be a valid integer	y- coordinate part of 'Y' must be an integer.
400	ValidationException	tile resolution part of 'Y' must be a valid integer followed by 'x'	Tile resolution part of 'Y' must be an integer followed by 'X'.

Codice di errore	Eccezione	Vecchio messaggio di errore	Nuovo messaggio di errore
400	ValidationException	file type extension part of 'Y' must not be empty if a '.' is present	File type extension part of 'Y' must not be empty if a '.' is present.
400	ValidationException	'Z' must be a valid integer	'Z' must be an integer.
400	ValidationException	'X' must be a valid integer	'X' must be an integer.
400	ValidationException	'Z' must not be less than minimum zoom of style '<Style>' (<Minimum Value>)	'Z' must not be less than minimum zoom of style <Style> (<MinimumValue>).
400	ValidationException	'Z' must not be greater than maximum zoom of style '<Style>' (<Maximum Value>)	'Z' must not be greater than maximum zoom of style Style (<MaximumValue>).
400	ValidationException	'Z' value not supported	'Z' must be between 0 and 63.
400	ValidationException	tile resolution part of 'Y' must be omitted because '<Style>' is a vector style	Tile resolution part of 'Y' must be omitted for style <Style>.
400	ValidationException	tile resolution part of 'Y' must be at least 1	Tile resolution part of 'Y' must be at least 1.

Codice di errore	Eccezione	Vecchio messaggio di errore	Nuovo messaggio di errore
400	ValidationException	tile resolution part of 'Y' must not be greater than max resolution of style '<Style>' (<Maximum Resolution>)	Tile resolution part of 'Y' must not be greater than maximum resolution of style <Style> (max <MaxResolution>).
400	ValidationException	file type extension part of 'Y' must be one of <SupportedFileFormats> (or may be omitted) for style '<Style>'	File type extension part of 'Y' must be one of <SupportedFileFormats> (or may be omitted) for style <Style>.
400	ValidationException	file type extension part of 'Y' must be omitted for style '<Style>'	File type extension part of 'Y' must be omitted for style <Style>.
400	ValidationException	y-coordinate part of 'Y' must be an integer in the range $0..2^{\text{Zoom}} - 1$ ( $0..<MaxTileCoordinate>$ )	y-coordinate part of 'Y' must be an integer in the range $0..2^{\text{Zoom}} - 1$ ( $0..<MaxTileCoordinate>$ ).
400	ValidationException	'DataSource' must be one of: Here, Esri	'DataSource' must be one of Esri, Grab, Here.
400	ValidationException	unsupported price plan '<PricingPlan>'	'PricingPlan' must be set to RequestBasedUsage.
400	ValidationException	Invalid token	'NextToken' must be a valid token.

Codice di errore	Eccezione	Vecchio messaggio di errore	Nuovo messaggio di errore
400	ValidationException	Expired token	'NextToken' must not be expired.
400	ValidationException	Unsupported Map Style: <Style>	<Style> is not a supported map style. For more information about list of supported map styles, see <a href="https://docs.aws.amazon.com/location/latest/APIReference/API_MapConfiguration.html">https://docs.aws.amazon.com/location/latest/APIReference/API_MapConfiguration.html</a> .
402	ServiceQuotaExceededException	Map resources per account exceeded quota limits. For more info, see <a href="https://aws.amazon.com/premiumsupport/knowledge-center/manage-service-limits/">https://aws.amazon.com/premiumsupport/knowledge-center/manage-service-limits/</a>	Map resources have exceeded the quota per account per region. For more information, see <a href="https://aws.amazon.com/premiumsupport/knowledge-center/manage-service-limits/">https://aws.amazon.com/premiumsupport/knowledge-center/manage-service-limits/</a> .
409	ConflictException	Resource already exists	Map already exists: <MapName>.

## Localizzatori

### Localizzatori

Codice di errore	Eccezione	Vecchio messaggio di errore	Nuovo messaggio di errore
500	InternalServerErrorException	Internal Server Exception  internal server error  unable to retrieve point from the storage  unable to verify tracker  Error processing List request	Internal server error. Try again later.
404	ResourceNotFoundException	tracker not found: <TrackerName>  Tracker with name <TrackerName> was not found	Tracker not found: <TrackerName>.
404	ResourceNotFoundException	association not found: TrackerName <TrackerName>; and ConsumerArn <ConsumerArn >	Association between tracker <TrackerName> and consumer <ConsumerArn> is not found.
400	ValidationException	'ConsumerArn' must refer to a geofence collection resource	'ConsumerArn' must refer to a geofence collection resource.
400	ValidationException	'ConsumerArn' must refer to a resource	'ConsumerArn' must refer to a resource

Codice di errore	Eccezione	Vecchio messaggio di errore	Nuovo messaggio di errore
		in the same region as the tracker it is associated to	in the same region as the tracker it is associated with.
400	ValidationException	'ConsumerArn' must refer to a resource in the same AWS account as the tracker it is associated to	'ConsumerArn' must refer to a resource in the same AWS account as the tracker it is associated with.
400	ValidationException	'DataSource' must be one of: Here, Esri	'DataSource' must be one of Esri, Grab, Here.
400	ValidationException	Nothing to update.	At least one of the following fields must be set: 'Description', 'PositionFiltering'
400	ValidationException	Invalid token	'NextToken' must be a valid token.
400	ValidationException	Expired token	'NextToken' must not be expired.
400	ValidationException	request.TrackerName not found on request	'TrackerName ' is a required field.
400	ValidationException	no deviceId parameter found	'DeviceId' is a required field.
400	ValidationException	unsupported price plan '<PricingPlan>'	'PricingPlan' must be set to RequestBasedUsage

Codice di errore	Eccezione	Vecchio messaggio di errore	Nuovo messaggio di errore
400	ValidationException	Invalid token	'NextToken' must be a valid token.
400	ValidationException	Expired token	'NextToken' must not be expired.
400	ValidationException	provided start time is incorrect, should follow the format YYYY-MM-DDThh:mm:ss.sssZ“	'StartTimeInclusive' must follow the format YYYY-MM-DDThh:mm:ss.sssZ.
400	ValidationException	provided end time is incorrect, should follow the format YYYY-MM-DDThh:mm:ss.sssZ	'EndTimeExclusive' must follow the format YYYY-MM-DDThh:mm:ss.sssZ.
400	ValidationException	end time must be after start time	'EndTimeExclusive' must be after 'StartTimeInclusive'.
400	ValidationException	invalid key state	KMS key must be a symmetric Customer Master Key (CMK). Invalid state found. For more information about how key state affects the use of a KMS key, see <a href="https://docs.aws.amazon.com/kms/latest/developerguide/key-state.html">https://docs.aws.amazon.com/kms/latest/developerguide/key-state.html</a> .



Codice di errore	Eccezione	Vecchio messaggio di errore	Nuovo messaggio di errore
400	ValidationException	key not found	Invalid KMS key. '<KmsKeyId>' <KmsKeyIdValue> not found.
400	ValidationException	key is disabled	Symmetric Customer Master Key (CMK) must be enabled.
400	ValidationException	access denied	Symmetric Customer Master Key (CMK) must allow Amazon Location to create grants to its KMS key.
402	ServiceQuotaExceededException	Tracker <TrackerName> may not have more than <Max> consumer associations	Tracker resource may not have more than <Max> consumer associations. For more information, see <a href="https://aws.amazon.com/premiumsupport/knowledge-center/manage-service-limits/">https://aws.amazon.com/premiumsupport/knowledge-center/manage-service-limits/</a> .

Codice di errore	Eccezione	Vecchio messaggio di errore	Nuovo messaggio di errore
402	ServiceQuotaExceededException	Trackers per account exceeded quota limits. For more info, see <a href="https://aws.amazon.com/premiumsupport/knowledge-center/manage-service-limits/">https://aws.amazon.com/premiumsupport/knowledge-center/manage-service-limits/</a>	Tracking resources have exceeded the quota per account per region. For more information, see <a href="https://aws.amazon.com/premiumsupport/knowledge-center/manage-service-limits/">https://aws.amazon.com/premiumsupport/knowledge-center/manage-service-limits/</a> .
409	ConflictException	association already exists: TrackerName <TrackerName>; and ConsumerArn <ConsumerArn>	An association already exists between tracker <TrackerName> and consumer <ConsumerArn>.
409	ConflictException	Tracker already exists: <TrackerName>	Tracker already exists: <TrackerName>.

## Route

### Route

Codice di errore	Eccezione	Vecchio messaggio di errore	Nuovo messaggio di errore
500	InternalServerErrorException	Internal Server Exception	Internal server error. Try again later.

Codice di errore	Eccezione	Vecchio messaggio di errore	Nuovo messaggio di errore
404	ResourceNotFoundException	Resource not found	Route calculator not found: <RouteCalculatorName>.
400	ValidationException	Invalid token	'NextToken' must be a valid token.
400	ValidationException	Expired token	'NextToken' must not be expired.
400	ValidationException	'DataSource' must be one of: Here, Esri, Grab	'DataSource' must be one of Esri, Grab, Here.
400	ValidationException	<PricingPlan> pricing plan is not supported	'PricingPlan' must be set to RequestBasedUsage
400	ValidationException	unsupported price plan '<PricingPlan>'	'PricingPlan' must be set to RequestBasedUsage
400	ValidationException	Grab is only supported in the ap-southeast-1 region	'DataSource' <DataSourceName> must only be used in following regions: ap-southeast-1.
400	ValidationException	PricingPlan must be 'RequestBasedUsage'	'PricingPlan' must be set to RequestBasedUsage.
400	ValidationException	'DeparturePositions[0][0]' must be between -180 and 180	'DeparturePositions[0][0]' must be between -180 and 180.

Codice di errore	Eccezione	Vecchio messaggio di errore	Nuovo messaggio di errore
400	ValidationException	'DeparturePositions[0][1]' must be between -90 and 90	'DeparturePositions[0][1]' must be between -90 and 90.
400	ValidationException	'DestinationPositions[0][0]' must be between -180 and 180	'DestinationPositions[0][0]' must be between -180 and 180.
400	ValidationException	'DestinationPositions[0][1]' must be between -90 and 90.	'DestinationPositions[0][1]' must be between -90 and 90
400	ValidationException	'DepartNow' may not be true if 'DepartureTime' is set	Only one of 'DepartNow' or 'DepartureTime' may be set.
400	ValidationException	'<TravelModeOption>' may not be set when 'TravelMode' has value <TravelModeOption>	'<TravelModeOption>' must not be set when 'TravelMode' has value <TravelModeOption>.
400	ValidationException	'CarModeOptions' may not be set when 'TravelMode' has value Walking	'CarModeOptions' must not be set when 'TravelMode' has value Walking.
400	ValidationException	'TruckModeOptions' may not be set when 'TravelMode' has value Walking	'TruckModeOptions' must not be set when 'TravelMode' has value Walking.

Codice di errore	Eccezione	Vecchio messaggio di errore	Nuovo messaggio di errore
400	ValidationException	'TruckModeOptions' may not be set when 'TravelMode' has value Car	'TruckModeOptions' must not be set when 'TravelMode' has value Car.
400	ValidationException	'CarModeOptions' may not be set when 'TravelMode' has value Truck	'CarModeOptions' must not be set when 'TravelMode' has value Truck.
400	ValidationException	At least one of [Height, Length, Width] must be set in 'TruckModeOptions.Dimensions'	At least one of the following attribute must be set in TruckModeOptions.Dimensions: Height, Length, Width.
400	ValidationException	At least one of [Total] must be set in 'TruckModeOptions.Weight'	At least one of the following attribute must be set in TruckModeOptions.Weight: Total.
400	ValidationException	'DeparturePositions' count must be 10 or less with DataSource set to Esri	'DeparturePositions' must have length at most 10 for 'DataSource' Esri.
400	ValidationException	'DestinationPositions' count must be 10 or less with DataSource set to Esri	'DestinationPositions' must have length at most 10 for 'DataSource' Esri.

Codice di errore	Eccezione	Vecchio messaggio di errore	Nuovo messaggio di errore
400	ValidationException	'DeparturePositions[0]' is more than 40km away from 'DestinationPositions[0]'	'DeparturePositions[0]' must not be more than 40 km away from 'DestinationPositions[0]'.
400	ValidationException	'DeparturePositions[0]' is more than 400km away from 'DestinationPositions[0]'	'DeparturePositions[0]' must not be more than 400 km away from 'DestinationPositions[0]'.
400	ValidationException	DeparturePositions[0] is contained within an unsupported region. Korea is not supported for CalculateRouteMatrix with the provider Esri.	DeparturePositions[0] is located in Korea, which is not supported when using CalculateRouteMatrix with data provider Esri.
400	ValidationException	'<HereTruckDimension>' must be between <Min> and <Max> <Unit>	'HereTruckDimension' must be between <Min> and <Max> <Unit>.
400	ValidationException	'WaypointPositions[0][0]' must be between -180 and 180	'WaypointPositions[0][0]' must be between -180 and 180.
400	ValidationException	'WaypointPositions[0][1]' must be between -90 and 90	'WaypointPositions[0][1]' must be between -90 and 90.

Codice di errore	Eccezione	Vecchio messaggio di errore	Nuovo messaggio di errore
400	ValidationException	'WaypointPositions[1][0]' must be between -180 and 180	'WaypointPositions[1][0]' must be between -180 and 180.
400	ValidationException	'WaypointPositions[1][1]' must be between -90 and 90	'WaypointPositions[1][1]' must be between -90 and 90.
400	ValidationException	No road segment could be matched for one or more coordinates within a radius (1km)	One or more provided positions are more than 1 km from the nearest road segment.
400	ValidationException	Some positions in the request are unreachable	Some positions in the request are unreachable.
400	ValidationException	Total distance between all waypoints must be not be greater than 40km for DataSource Esri when using TravelMode Walking	Total distance between all route positions must not be greater than 40 km for 'DataSource' Esri and 'TravelMode' Walking.
400	ValidationException	Total distance between all waypoints must be not be greater than 400km for DataSource Esri	Total distance between all route positions must not be greater than 400 km for 'DataSource' Esri.

Codice di errore	Eccezione	Vecchio messaggio di errore	Nuovo messaggio di errore
400	ValidationException	Following positions in the request are unreachable: <UnreachablePositions>	The following positions are unreachable: <UnreachablePositions>.
400	ValidationException	'DepartureTime' contains a badly-formatted timestamp	'DepartureTime' must follow the format YYYY-MM-DDThh:mm:ss.sssZ.
400	ValidationException	'TravelMode' <TravelMode> is not supported by <DataProvider>	'TravelMode' <TravelMode> not supported by data provider <DataProvider>.
400	ValidationException	'DeparturePositions' must be set	'DeparturePositions' must not be empty.
400	ValidationException	'DestinationPositions' must be set	'DestinationPositions' must not be empty.
400	ValidationException	Some inputs in the request are invalid	Some inputs in the request are invalid.
400	ValidationException	No route found between position <FirstPosition> and position <SecondPosition>	No route found between position <FirstPosition> and position <SecondPosition>.



Codice di errore	Eccezione	Vecchio messaggio di errore	Nuovo messaggio di errore
400	ValidationException	No route found	No route found. For more information, see <a href="https://developer.amazon.com/documentation/routing-api/dev_guide/topics/no_route.html">https://developer.amazon.com/documentation/routing-api/dev_guide/topics/no_route.html</a> .
400	ValidationException	No route found	No route found.
402	ServiceQuotaExceededException	Route calculators per account exceeded quota limits. For more info, see <a href="https://aws.amazon.com/premiumsupport/knowledge-center/manage-service-limits/">https://aws.amazon.com/premiumsupport/knowledge-center/manage-service-limits/</a>	Route calculator resources have exceeded the quota per account per region. For more information, see <a href="https://aws.amazon.com/premiumsupport/knowledge-center/manage-service-limits/">https://aws.amazon.com/premiumsupport/knowledge-center/manage-service-limits/</a> .
409	ConflictException	Resource already exists	Route calculator already exists: <RouteCalculatorName>.

## Metadati

### Metadati

Codice di errore	Eccezione	Vecchio messaggio di errore	Nuovo messaggio di errore
500	InternalServerErrorException	Internal Server Error Error processing List request	Internal server error. Try again later.
404	ResourceNotFoundException	APIKey not found	Api key not found: <APIKeyName>.
404	ResourceNotFoundException	APIKeyID not found	ApiKeyId not found: <APIKeyID>.
400	ValidationException	Either ExpireTime or NoExpiry must be provided	At least one of the following fields must be set: 'ExpireTime', 'NoExpiry'.
400	ValidationException	NoExpiry cannot be set to false if no ExpireTime is provided	'ExpireTime' must be set when 'NoExpiry' has value false.
400	ValidationException	ExpireTime cannot be set if NoExpiry is true	'ExpireTime' must not be set when 'NoExpiry' has value true.
400	ValidationException	Expire time '<ExpireTimeValue>' is not a valid time format	'ExpireTime' must follow the format YYYY-MM-DDThh:mm:ss.sssZ.
400	ValidationException	Expire time '<ExpireTimeValue>' cannot	'ExpireTime' must not be in the past.

Codice di errore	Eccezione	Vecchio messaggio di errore	Nuovo messaggio di errore
		be in the past when creating a key	
400	ValidationException	Invalid token	'NextToken' must be a valid token.
400	ValidationException	Expired token	'NextToken' must not be expired.
400	ValidationException	The API Key %s has been recently used and the requested update may impact current usage. Specify ForceUpdate=true to update the API Key configuration.	This update may cause some users to lose API access. Because this API Key has been used in the last 7 days, you must set 'ForceUpdate' to true to confirm this change.
400	ValidationException	Expire time '<ExpireTimeValue>' must not be more than 1 minute in the past	'ExpireTime' must not be more than 1 minute in the past.
400	ValidationException	Description, ExpireTime, NoExpiry and Restrictions can't all be empty	At least one of the following fields must be set: 'Description', 'ExpireTime', 'NoExpiry', 'Restrictions'.
400	ValidationException	API Key expired	'ApiKeyId' must not be expired.

Codice di errore	Eccezione	Vecchio messaggio di errore	Nuovo messaggio di errore
409	ConflictException	API key named <APIKeyName> already exists	Api key already exists: <APIKeyName>.

## Geofence

### Geofenzioni

Codice di errore	Eccezione	Vecchio messaggio di errore	Nuovo messaggio di errore
500	InternalServerErrorException	<p>internal server error</p> <p>Internal server error</p> <p>Unsupported geofence geometry encountered</p> <p>geometry marshal error</p> <p>geometry load error</p> <p>unable to get geofence collection</p> <p>unable to delete geofences</p> <p>unable to retrieve geofence</p> <p>Error processing List request</p>	Internal server error. Try again later.

Codice di errore	Eccezione	Vecchio messaggio di errore	Nuovo messaggio di errore
404	ResourceNotFoundException	<p>collection not found: &lt;GeofenceCollectionName&gt;</p> <p>&lt;GeofenceCollectionName&gt; geofence collection not found</p> <p>Resource not found error</p> <p>no geofence with given name found</p>	Geofence Collection not found: <GeofenceCollectionName>.
400	ValidationException	unsupported price plan '<PricingPlan>'	'PricingPlan' must be set to RequestBasedUsage.
400	ValidationException	<p>KMS key must be a symmetric CMK.</p> <p>Invalid usage type: &lt;UsageType&gt;</p>	<p>KMS key must be a symmetric Customer Master Key (CMK).</p> <p>Invalid usage type &lt;UsageType&gt;. For how to create a symmetric CMK, refer to <a href="https://docs.aws.amazon.com/kms/latest/developerguide/create-keys.html#create-symmetric-cmk">https://docs.aws.amazon.com/kms/latest/developerguide/create-keys.html#create-symmetric-cmk</a>.</p>
400	ValidationException	Invalid token	'NextToken' must be a valid token.

Codice di errore	Eccezione	Vecchio messaggio di errore	Nuovo messaggio di errore
400	ValidationException	Expired token	'NextToken' must not be expired.
400	ValidationException	PricingPlanDataSource cannot be updated without updating PricingPlan	'PricingPlan' must be provided to update 'PricingPlanDataSource'.
400	ValidationException	nothing to update	At least one of the following fields must be set: 'Description'
400	ValidationException	invalid key state	KMS key must be a symmetric Customer Master Key (CMK). Invalid state <InvalidState>. For more information about how key state affects the use of a KMS key, see <a href="https://docs.aws.amazon.com/kms/latest/developerguide/key-state.html">https://docs.aws.amazon.com/kms/latest/developerguide/key-state.html</a> .
400	ValidationException	key not found	Invalid KMS key. '<KmsKeyId>' <KmsKeyIdValue> not found.
400	ValidationException	key is disabled	Symmetric Customer Master Key (CMK) must be enabled.

Codice di errore	Eccezione	Vecchio messaggio di errore	Nuovo messaggio di errore
400	ValidationException	access denied	Symmetric Customer Master Key (CMK) must allow Amazon Location to create grants to its KMS key.
400	ValidationException	duplicate geofence ID in batch	'GeofenceId' <DuplicatedGeofenceId> is duplicated in batch.
400	ValidationException	missing GeofenceId	'GeofenceId' must not be empty.
400	ValidationException	Invalid token	'NextToken' must be a valid token.
400	ValidationException	Expired token	'NextToken' must not be expired.
400	ValidationException	Position[0] must be between -180 and 180	'Position[0]' must be between -180 and 180.
400	ValidationException	Position[1] must be between -90 and 90	'Position[1]' must be between -90 and 90.
400	ValidationException	radius must be less than or equal to 1000km	'Geometry.Circle.Radius' must be less than or equal to 1000km.
400	ValidationException	no geofence with given name found	Geofence not found: <CollectionName>.

Codice di errore	Eccezione	Vecchio messaggio di errore	Nuovo messaggio di errore
400	ValidationException	Geometry must contain either a Circle or Polygon, not both	Only one of 'Circle' or 'Polygon' may be set within 'Geometry'.
400	ValidationException	Geometry must contain a Polygon or a Circle	One of 'Polygon' or 'Circle' must be set within 'Geometry'.
400	ValidationException	radius must be greater than 0m	'Geometry.Circle.Radius' must be greater than 0m.
400	ValidationException	empty polygon	'Geometry.Polygon' must not be empty.
400	ValidationException	empty polygon ring	'Geometry.Polygon' must not be empty.
400	ValidationException	circle can not cross antimeridian	'Geometry.Circle' must not cross antimeridian. Cut it in two such that neither part's representation crosses the antimeridian.
400	ValidationException	polygon can not cross antimeridian	'Geometry.Polygon' must not cross antimeridian. Cut it in two such that neither part's representation crosses the antimeridian.



Codice di errore	Eccezione	Vecchio messaggio di errore	Nuovo messaggio di errore
400	ValidationException	polygon can not have interior rings (holes), remove holes	'Geometry.Polygon' must not have interior rings (holes). For more information about interior rings see <a href="https://www.rfc-editor.org/rfc/rfc7946.html#appendix-A.3">https://www.rfc-editor.org/rfc/rfc7946.html#appendix-A.3</a> .
400	ValidationException	polygon ring is not closed	'Geometry.Polygon' contains an open ring. Close the ring by ensuring the first and last positions are equal.
400	ValidationException	polygon ring has more than 1000 vertices	'Geometry.Polygon' must not have more than 1000 vertices.
400	ValidationException	polygon ring has fewer than 4 positions	Number of vertices in 'Geometry.Polygon' must be greater or equal to 4.
400	ValidationException	invalid center	'Geometry.Circle.Center' must be a valid position (longitude/latitude pair).
400	ValidationException	radius must be greater than 0m	'Geometry.Circle.Radius' must be greater than 0 m.

Codice di errore	Eccezione	Vecchio messaggio di errore	Nuovo messaggio di errore
400	ValidationException	longitude range should be between -180 and 180 degrees	Longitude must be between -180 and 180 degrees, but was set to <Provided Longitude>.
400	ValidationException	latitude range should be between -90 and 90 degrees	Latitude must be between -90 and 90 degrees, but was set to <Provided Longitude>.
400	ValidationException	polygon exterior ring is expected to be counter clockwise	'Geometry.Polygon' must be oriented counter-clockwise.
400	ValidationException	polygon interior ring should be clockwise oriented	'Geometry.Polygon' must be oriented clockwise.
400	ValidationException	radius must be less than or equal to 1000km	'Geometry.Circle.R adius' must be less than or equal to 1000 km.
400	ValidationException	timestamp.Parse() error	'SampleTime' must follow the format YYYY-MM-DDThh:mm:ss.sssZ.
400	ValidationException	invalid input	'SourceArn' must refer to a tracker resource.

Codice di errore	Eccezione	Vecchio messaggio di errore	Nuovo messaggio di errore
400	ValidationException	arn: invalid prefix	'SourceArn' must be a valid ARN. For more information, see <a href="https://docs.aws.amazon.com/general/latest/gr/AWS-arns-and-namespaces.html">https://docs.aws.amazon.com/general/latest/gr/AWS-arns-and-namespaces.html</a> .
400	ValidationException	arn: not enough sections	'SourceArn' must be a valid ARN. For more information, see <a href="https://docs.aws.amazon.com/general/latest/gr/AWS-arns-and-namespaces.html">https://docs.aws.amazon.com/general/latest/gr/AWS-arns-and-namespaces.html</a> .
400	ValidationException	invalid resource part	'SourceArn' must refer to a tracker resource.
402	ServiceQuotaExceededException	Geofence collections per account exceeded quota limits. For more info, see <a href="https://aws.amazon.com/premiumsupport/knowledge-center/manage-service-limits/">https://aws.amazon.com/premiumsupport/knowledge-center/manage-service-limits/</a>	Geofence collection resources have exceeded the quota per account per region. For more information, see <a href="https://aws.amazon.com/premiumsupport/knowledge-center/manage-service-limits/">https://aws.amazon.com/premiumsupport/knowledge-center/manage-service-limits/</a> .

Codice di errore	Eccezione	Vecchio messaggio di errore	Nuovo messaggio di errore
409	ConflictException	collection already exists: <Geofence CollectionName>	Geofence Collection already exists: <GeofenceCollectionName>.
409	ConflictException	Resource conflict error	Geofence already exists: <Geofence Name>.

## Esempi di codice e tutorial per lavorare con Amazon Location Service

Questo argomento mostra un elenco di esempi di codice, tutorial e post di blog per aiutarti a conoscere Amazon Location Service. Ogni esempio di codice include una descrizione di come funziona.

[Puoi trovare altri esempi nella pagina AWS Geospatial, GitHub nella pagina degli AWS esempi GitHub per Amazon Location e sul sito del AWS blog.](#)

### Note

È utile comprendere la differenza tra la pagina AWS Geospatial GitHub e la pagina degli esempi. AWS GitHub

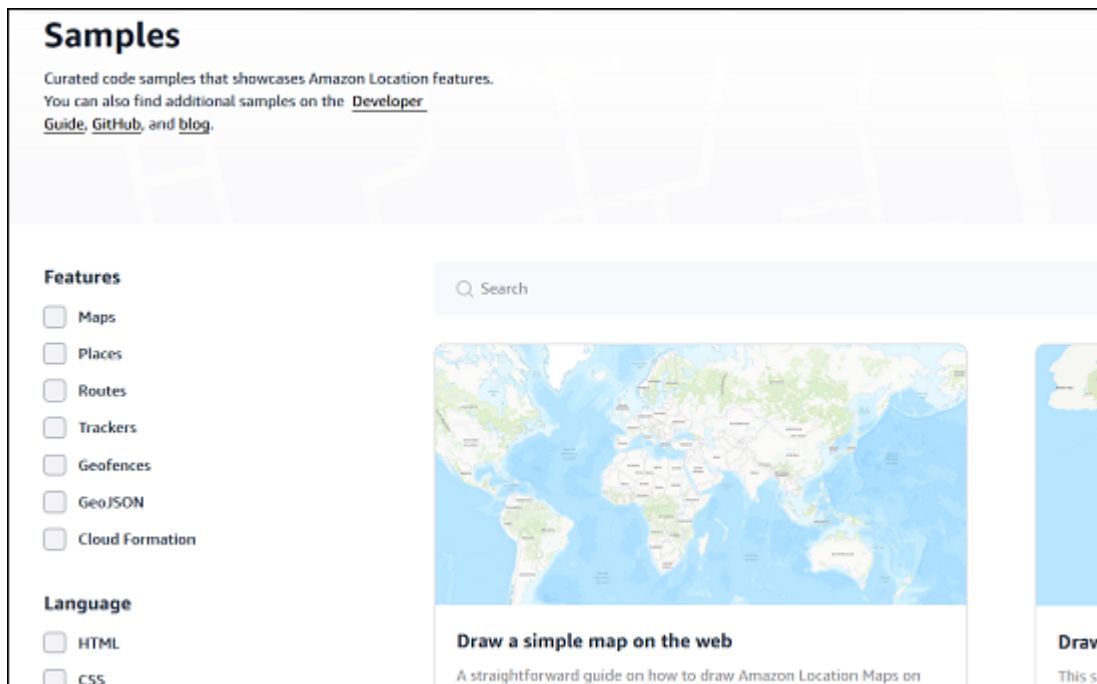
- Geospatial GitHub: la [GitHub pagina AWS Geospatial](#) include esempi creati e gestiti dal team di Amazon Location Service.
- Esempi GitHub: la [GitHub pagina degli AWS esempi per Amazon Location](#) include esempi che sono stati creati per Amazon Location, ma che possono o meno essere gestiti attivamente.

Il tutorial di [avvio rapido](#) è un buon punto di partenza prima di utilizzare altri esempi, in quanto mostra come completare i prerequisiti utili per la maggior parte degli esempi.

## Argomenti

- [Sito demo di Amazon Location](#)
- [Tutorial: avvio rapido](#)
- [Tutorial: arricchimento del database](#)
- [Esempio: app Explore](#)
- [Esempio: assegnare uno stile a una mappa](#)
- [Esempio: disegna dei marcatori](#)
- [Esempio: disegna punti raggruppati](#)
- [Esempio: disegnare un poligono](#)
- [Esempio: modifica della lingua della mappa](#)
- [Blog: notifiche sui tempi di consegna stimati](#)
- [Esempio: aggiornamenti della posizione dello stream](#)
- [Esempio: applicazione mobile di geofencing e tracciamento](#)

## Sito demo di Amazon Location



**Samples**

Curated code samples that showcases Amazon Location features. You can also find additional samples on the [Developer Guide](#), [GitHub](#), and [blog](#).

**Features**

- Maps
- Places
- Routes
- Trackers
- Geofences
- GeoJSON
- Cloud Formation

**Language**

- HTML
- CSS

Search

**Draw a simple map on the web**

A straightforward guide on how to draw Amazon Location Maps on

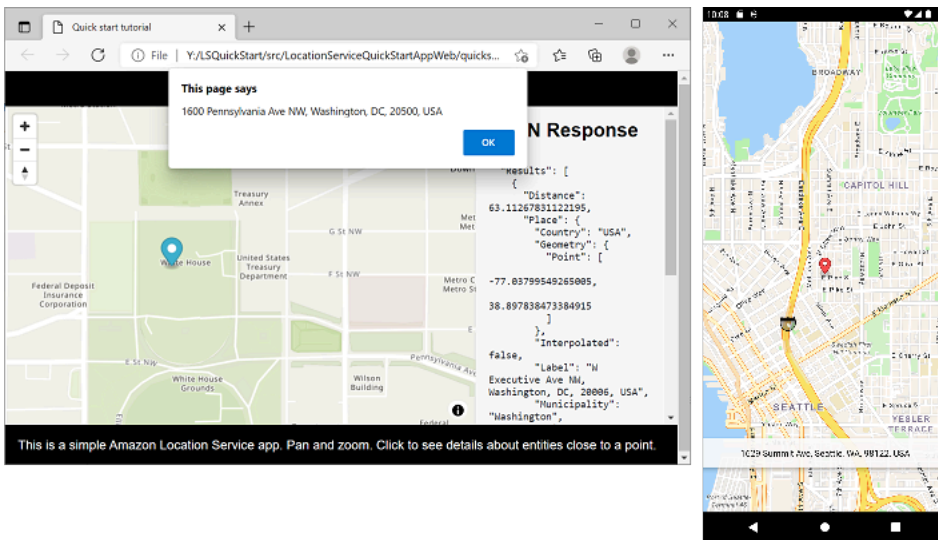
**Draw**

This sa

Puoi vedere le demo con il codice sorgente di Amazon Location Service in azione sul [sito Amazon Location Demo](#). Questo sito include una [demo web ospitata](#) e anche un'app demo per [Android](#).

Puoi anche trovare un'ampia gamma di esempi, filtrabili per funzionalità, lingua e piattaforma nella pagina [Esempi](#) del sito.

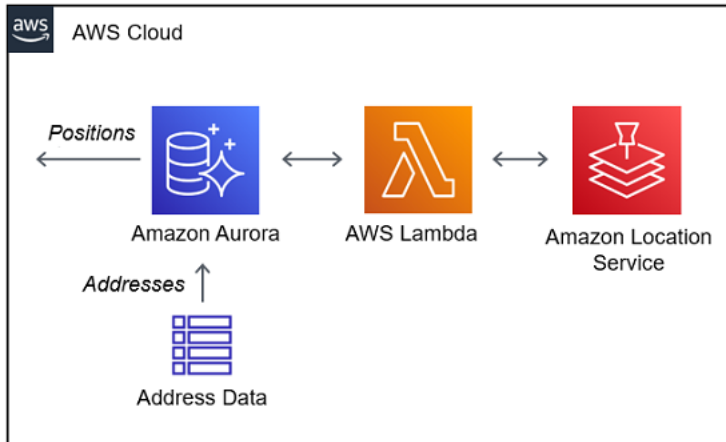
## Tutorial: avvio rapido



Sono disponibili tutorial di avvio rapido per dispositivi Web, iOS e Android. Per ogni piattaforma, il tutorial mostra come aggiungere una mappa interattiva a un'applicazione e come effettuare chiamate alle API di Amazon Location Service dalla tua applicazione. Il tutorial è disponibile per JavaScript una pagina Web statica, Kotlin per un'applicazione per telefoni Android o Swift per un'applicazione iOS.

- JavaScript per un link statico alla documentazione di una pagina web: [Creazione di un'app Web](#)
- Link alla documentazione di Kotlin per un'applicazione Android: [Avvio rapido con Amazon Location Service](#)
- Link alla documentazione di Swift per un'app iOS: [Creazione di un'app per iOS](#)

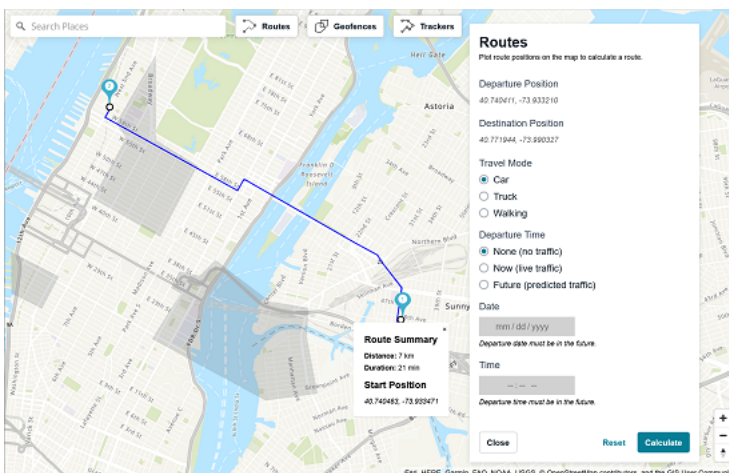
## Tutorial: arricchimento del database



Questo tutorial mostra come usare Amazon Location Service, chiamato da AWS Lambda per normalizzare gli indirizzi e aggiungere latitudine e longitudine ai record in un database Amazon Aurora. Utilizza Amazon Aurora e AWS Lambda

Link alla documentazione: [Amazon Aurora PostgreSQL funzioni definite dall'utente per Amazon Location Service](#)

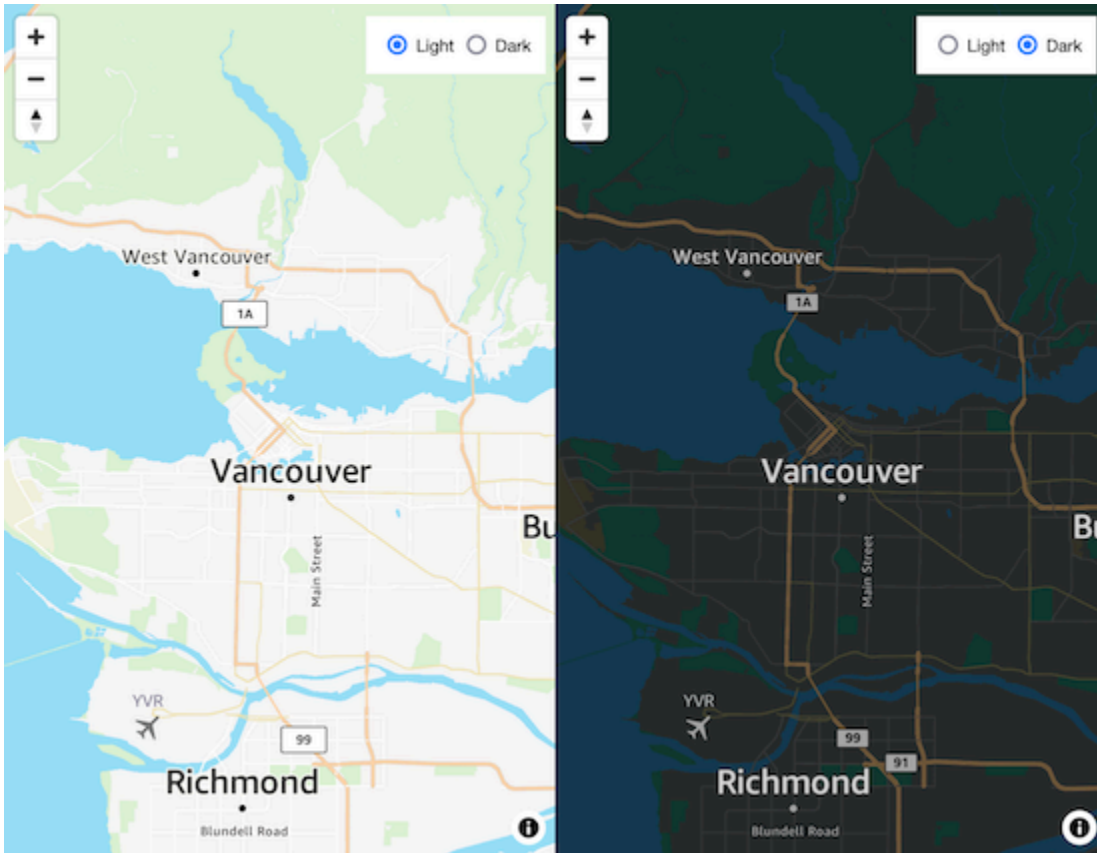
## Esempio: app Explore



Uno dei modi migliori per conoscere le funzionalità di Amazon Location Service è utilizzare la [funzionalità Explore](#) all'interno della console Amazon Location. Questo esempio completo di applicazione web imita le funzionalità di mappe, luoghi, percorsi, geofence e tracker della console per mostrarti come ricreare queste funzionalità nella tua app. Utilizza Amplify, React e JavaScript

GitHub Link agli esempi: [Esplora](#) l'applicazione di esempio

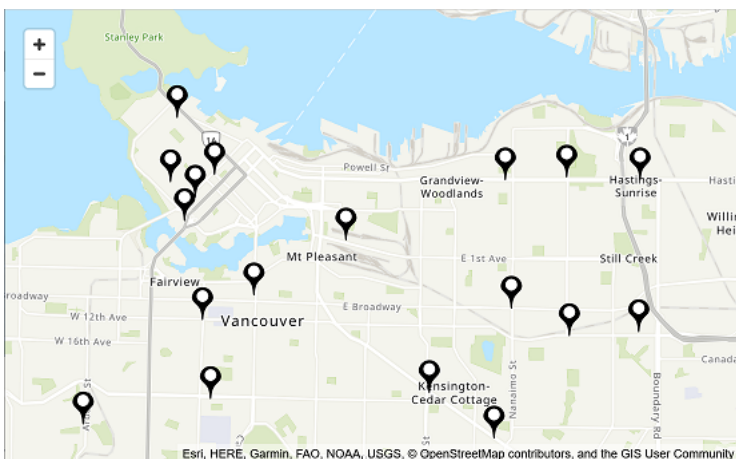
## Esempio: assegnare uno stile a una mappa



Questo esempio di codice mostra come passare da una mappa satellitare a una mappa stradale vettoriale, utilizzando MapLibre in JavaScript. Utilizza MapLibre, l'helper di autenticazione Amazon Location e JavaScript.

GitHub Collegamento geospaziale: mappa [interattiva](#) con cambio di stile

## Esempio: disegna dei marcatori

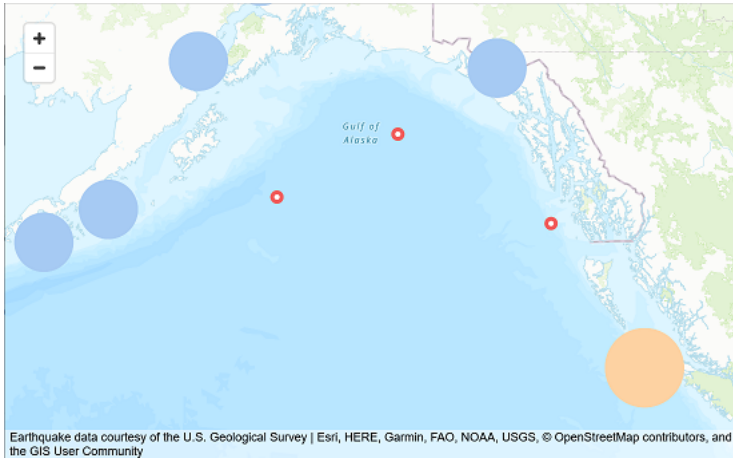




Questo esempio di codice mostra le sedi di Amazon Locker a Vancouver, BC, Canada. Mostra come disegnare marcatori nelle posizioni dei punti. Utilizza Node.js MapLibre, React, l'helper di autenticazione Amazon Location e JavaScript.

GitHub Collegamento geospaziale: [mappa interattiva](#) con indicatori in punti

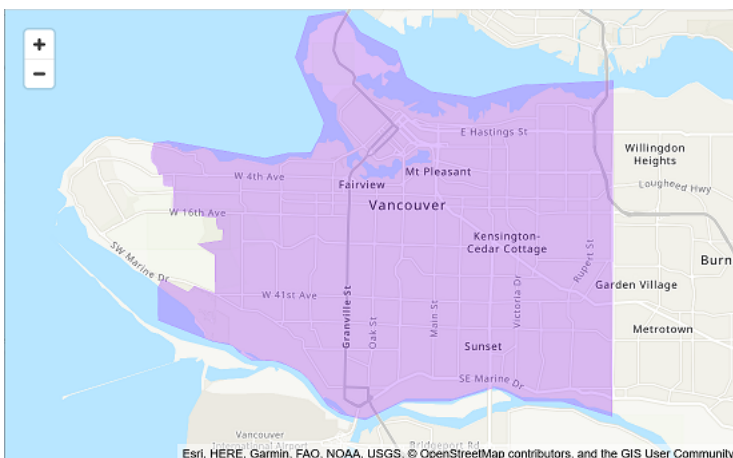
## Esempio: disegna punti raggruppati



Utilizzando i dati sui terremoti USGS, questo esempio di codice mostra come disegnare punti che si raggruppano quando sono vicini tra loro sulla mappa. Uses MapLibre, Node.js, React, Amplify e. JavaScript

GitHub Link agli esempi: [Mappa interattiva con gruppi](#) di punti

## Esempio: disegnare un poligono



Questo esempio di codice mostra come disegnare un poligono sulla mappa. Utilizza Node.js MapLibre, React, l'helper di autenticazione Amazon Location e JavaScript.

GitHub [Collegamento geospaziale: mappa interattiva con poligoni](#)

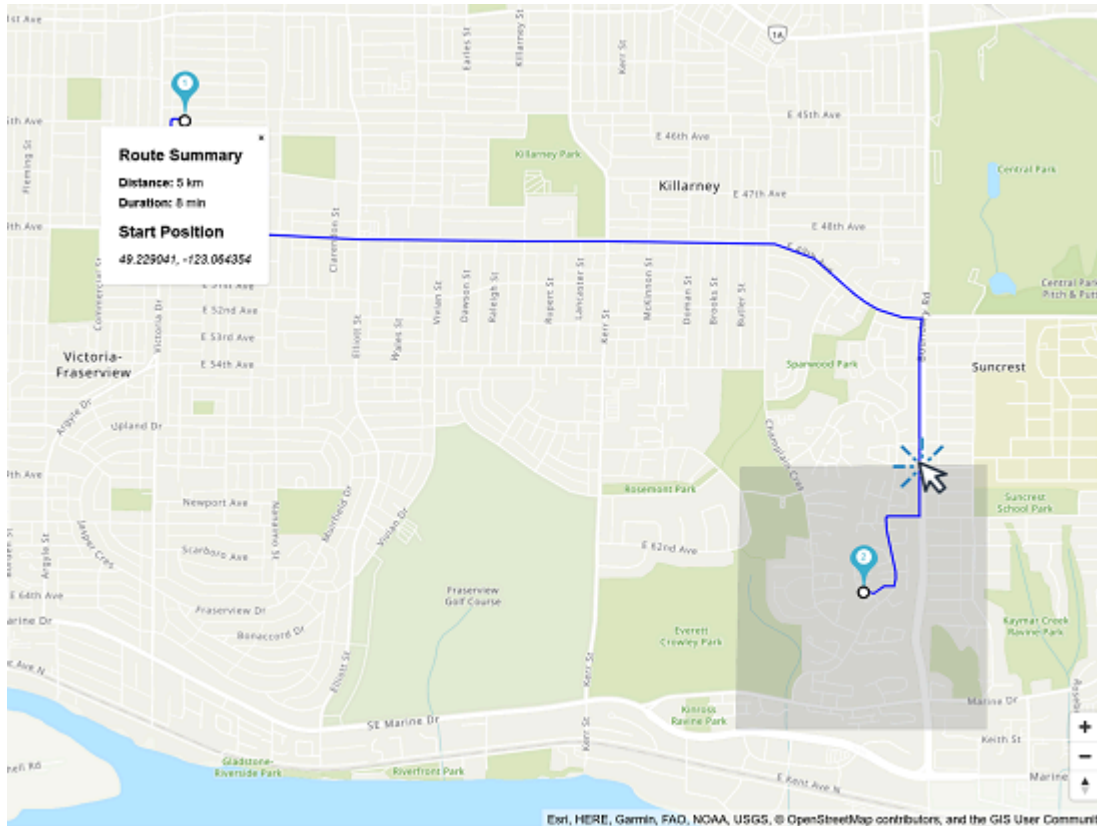
## Esempio: modifica della lingua della mappa



Questo esempio di codice mostra come modificare la lingua di visualizzazione delle mappe in Amazon Location. Utilizza Amplify, React e MapLibre

GitHub Link agli esempi: Esempio di [modifica della lingua della mappa](#)

## Blog: notifiche sui tempi di consegna stimati



Questo post sul blog mostra diversi modi per notificare ai clienti i tempi di consegna stimati. Spiega come utilizzare i percorsi per mostrare i tempi di guida stimati e quindi utilizzare tracker e geofence per avvisare quando un conducente si avvicina al cliente. Utilizza Amplify, React EventBridge, Amazon e Amazon Simple Notification Service (Amazon SNS).

Link al blog: [Ora di arrivo prevista e notifiche di prossimità](#)

## Esempio: aggiornamenti della posizione dello stream



App Kinesis Stream To Tracker: questo esempio dimostra come utilizzare Kinesis Data Stream per pubblicare gli aggiornamenti dei tracker con Amazon Location Service. L'esempio è un'applicazione lambda distribuibile scritta in python che può essere integrata con un Kinesis Data Stream per utilizzare gli eventi Kinesis e aggiornare in batch le posizioni dei dispositivi.

Link al repository: [Amazon Location Amazon Kinesis Data Streams Stream To Tracker App](#)

[Per ulteriori informazioni sul tracciamento e sui geofence, consulta la documentazione Geofences and Trackers.](#) [Gli sviluppatori possono distribuire l'app seguendo la documentazione di AWS Serverless Application Repository o direttamente dalla console Lambda.](#)

App di esempio per lo streaming della posizione del dispositivo: questo esempio di codice mostra come trasmettere i dati sulla posizione del dispositivo a un Kinesis Data Stream e come funzionano le notifiche geofence. Questa app dipende dall'esecuzione dell'app di esempio Kinesis Stream to Tracker, elencata sopra, affinché le posizioni del tracker in streaming vengano aggiornate in Amazon Location Service.

Link al repository: app di [esempio di Amazon Location Device Position Streaming](#)

## Esempio: applicazione mobile di geofencing e tracciamento

Questa applicazione di esempio mostra come interagiscono un tracker e un geofence utilizzando una combinazione di funzionalità Lambda AWS IoT e Amazon Location. Sono disponibili tutorial per iOS e Android.

Link al tutorial: [esempio di applicazione mobile Geofence and Tracker](#)

# Come usare Amazon Location Service

Puoi utilizzare le funzionalità di Amazon Location Service per completare attività geografiche e relative alla posizione. Puoi quindi combinare queste attività per affrontare casi d'uso più complessi come il geomarketing, la consegna e il monitoraggio delle risorse.

Quando sei pronto a integrare funzionalità di localizzazione nella tua applicazione, utilizza i seguenti metodi per utilizzare la funzionalità Amazon Location Service, a seconda dei tuoi obiettivi e delle tue inclinazioni:

- **Strumenti di esplorazione:** se desideri sperimentare con le risorse di Amazon Location, i seguenti strumenti sono il modo più veloce per accedere e provare le API:
  - La [console Amazon Location](#) offre una varietà di strumenti di accesso rapido. Puoi creare e gestire le tue risorse e provare le API utilizzando [la](#) pagina Esplora. La console è utile anche per creare risorse (in genere un'attività singola) in preparazione all'utilizzo di uno qualsiasi degli altri metodi descritti in seguito.
  - L'[interfaccia a riga di AWS comando](#) (CLI) consente di creare risorse e accedere alle API di Amazon Location utilizzando un terminale. AWS CLIGestisce l'autenticazione quando la configuri con le tue credenziali.
  - Puoi vedere [esempi di codice e tutorial](#) che mostrano come eseguire attività utilizzando le API di Amazon Location Service. Ciò include [un esempio](#) che imita gran parte delle funzionalità della pagina Esplora nella console.
- **SDK della piattaforma:** se non stai visualizzando i dati su una mappa, puoi utilizzare uno qualsiasi degli [strumenti AWS standard](#) su cui basarti. AWS
  - Sono disponibili i seguenti SDK: C++, Go, Java, .NET JavaScript, Node.js, PHP, Python e Ruby.
- **SDK e librerie di frontend:** se desideri utilizzare Amazon Location per creare un'applicazione su una piattaforma mobile o visualizzare i dati su una mappa su qualsiasi piattaforma, hai le seguenti opzioni:
  - Le AWS Amplify librerie integrano Amazon Location all'interno di applicazioni [iOS](#), [Android](#) e [JavaScript](#)Web.
  - Le MapLibre librerie consentono di eseguire il rendering di mappe lato client in applicazioni [iOS](#), [Android](#) e [JavaScript](#)Web utilizzando Amazon Location.
  - [Le librerie Tangram ES consentono di renderizzare mappe 2D e 3D da dati vettoriali utilizzando OpenGL ES all'interno di applicazioni Web iOS e Android.](#) C'è anche Tangram per le applicazioni web. [JavaScript](#)

- Invio di richieste HTTPS dirette: se lavori con un linguaggio di programmazione per il quale non è disponibile un SDK o se desideri un maggiore controllo su come inviare una richiesta a AWS, puoi accedere ad Amazon Location inviando richieste HTTPS dirette autenticate dal processo di firma Signature Version 4. Per ulteriori informazioni sul [processo di firma di Signature Version 4](#), consulta la. Riferimenti generali di AWS

Questo capitolo descrive molte delle attività comuni alle applicazioni che utilizzano dati sulla posizione. La sezione relativa ai [casi d'uso comuni](#) descrive come combinarli con altri AWS servizi per ottenere casi d'uso più complessi.

## Argomenti

- [Prerequisiti per l'utilizzo di Amazon Location Service](#)
- [Utilizzo di Amazon Location Maps nella tua applicazione](#)
- [Ricerca di dati relativi a località e geolocalizzazione utilizzando Amazon Location](#)
- [Calcolo dei percorsi utilizzando Amazon Location Service](#)
- [Geofencing di un'area di interesse utilizzando Amazon Location](#)
- [Taggare le risorse di Amazon Location Service](#)
- [Concessione dell'accesso ad Amazon Location Service](#)
- [Monitoraggio di Amazon Location Service](#)
- [Creazione di risorse Amazon Location Service con AWS CloudFormation](#)

## Prerequisiti per l'utilizzo di Amazon Location Service

Questa sezione descrive cosa devi fare per usare Amazon Location Service. È necessario disporre Account AWS e configurare l'accesso ad Amazon Location per gli utenti che desiderano utilizzarlo.

## Registrati per un Account AWS

Se non ne hai uno Account AWS, completa i seguenti passaggi per crearne uno.

Per iscriverti a un Account AWS

1. Apri la pagina all'indirizzo <https://portal.aws.amazon.com/billing/signup>.
2. Segui le istruzioni online.

Nel corso della procedura di registrazione riceverai una telefonata, durante la quale sarà necessario inserire un codice di verifica attraverso la tastiera del telefono.

Quando ti iscrivi a un Account AWS, Utente root dell'account AWS viene creato un. L'utente root dispone dell'accesso a tutte le risorse e tutti i Servizi AWS nell'account. Come procedura consigliata in materia di sicurezza, assegna l'accesso amministrativo a un utente e utilizza solo l'utente root per eseguire [attività che richiedono l'accesso da parte dell'utente root](#).

AWS ti invia un'e-mail di conferma dopo il completamento della procedura di registrazione. È possibile visualizzare l'attività corrente dell'account e gestire l'account in qualsiasi momento accedendo all'indirizzo <https://aws.amazon.com/> e selezionando Il mio account.

## Crea un utente con accesso amministrativo

Dopo esserti registrato Account AWS, proteggi Utente root dell'account AWS AWS IAM Identity Center, abilita e crea un utente amministrativo in modo da non utilizzare l'utente root per le attività quotidiane.

Proteggi i tuoi Utente root dell'account AWS

1. Accedi [AWS Management Console](#) come proprietario dell'account scegliendo Utente root e inserendo il tuo indirizzo Account AWS email. Nella pagina successiva, inserisci la password.

Per informazioni sull'accesso utilizzando un utente root, consulta la pagina [Signing in as the root user](#) della Guida per l'utente di Accedi ad AWS .

2. Abilita l'autenticazione a più fattori (MFA) per l'utente root.

Per istruzioni, consulta [Abilitare un dispositivo MFA virtuale per l'utente Account AWS root \(console\)](#) nella Guida per l'utente IAM.

Crea un utente con accesso amministrativo

1. Abilita Centro identità IAM.

Per istruzioni, consulta [Abilitazione di AWS IAM Identity Center](#) nella Guida per l'utente di AWS IAM Identity Center .

2. In IAM Identity Center, concedi l'accesso amministrativo a un utente.



Per un tutorial sull'utilizzo di IAM Identity Center directory come fonte di identità, consulta [Configurare l'accesso utente con le impostazioni predefinite IAM Identity Center directory](#) nella Guida per l'AWS IAM Identity Center utente.

### Accedi come utente con accesso amministrativo

- Per accedere con l'utente IAM Identity Center, utilizza l'URL di accesso che è stato inviato al tuo indirizzo e-mail quando hai creato l'utente IAM Identity Center.

Per informazioni sull'accesso utilizzando un utente IAM Identity Center, consulta [AWS Accedere al portale di accesso](#) nella Guida per l'Accedi ad AWS utente.

### Assegna l'accesso ad altri utenti

1. In IAM Identity Center, crea un set di autorizzazioni che segua la migliore pratica di applicazione delle autorizzazioni con privilegi minimi.

Per istruzioni, consulta [Creare un set di autorizzazioni](#) nella Guida per l'utente.AWS IAM Identity Center

2. Assegna gli utenti a un gruppo, quindi assegna l'accesso Single Sign-On al gruppo.

Per istruzioni, consulta [Aggiungere gruppi](#) nella Guida per l'utente.AWS IAM Identity Center

## Concedi l'accesso ad Amazon Location Service

Per impostazione predefinita, i tuoi utenti non amministratori non dispongono di autorizzazioni. Prima che possano accedere ad Amazon Location, devi concedere l'autorizzazione allegando una policy IAM con autorizzazioni specifiche. Assicurati di seguire il principio del privilegio minimo quando concedi l'accesso alle risorse.

### Note

Per informazioni su come concedere agli utenti non autenticati l'accesso alla funzionalità di Amazon Location Service (ad esempio, in un'applicazione basata sul Web), consulta [Concessione dell'accesso ad Amazon Location Service](#)

La seguente policy di esempio fornisce all'utente l'autorizzazione ad accedere a tutte le operazioni di Amazon Location. Per ulteriori esempi, consulta [Esempi di policy basate sull'identità per Amazon Location Service](#).

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "geo:*"
      ],
      "Resource": "*",
      "Effect": "Allow"
    }
  ]
}
```

Per fornire l'accesso, aggiungi autorizzazioni ai tuoi utenti, gruppi o ruoli:

- Utenti e gruppi in AWS IAM Identity Center:

Crea un set di autorizzazioni. Segui le istruzioni riportate nella pagina [Create a permission set](#) (Creazione di un set di autorizzazioni) nella Guida per l'utente di AWS IAM Identity Center .

- Utenti gestiti in IAM tramite un provider di identità:

Crea un ruolo per la federazione delle identità. Segui le istruzioni riportate nella pagina [Creating a role for a third-party identity provider \(federation\)](#) (Creazione di un ruolo per un provider di identità di terze parti [federazione]) nella Guida per l'utente di IAM.

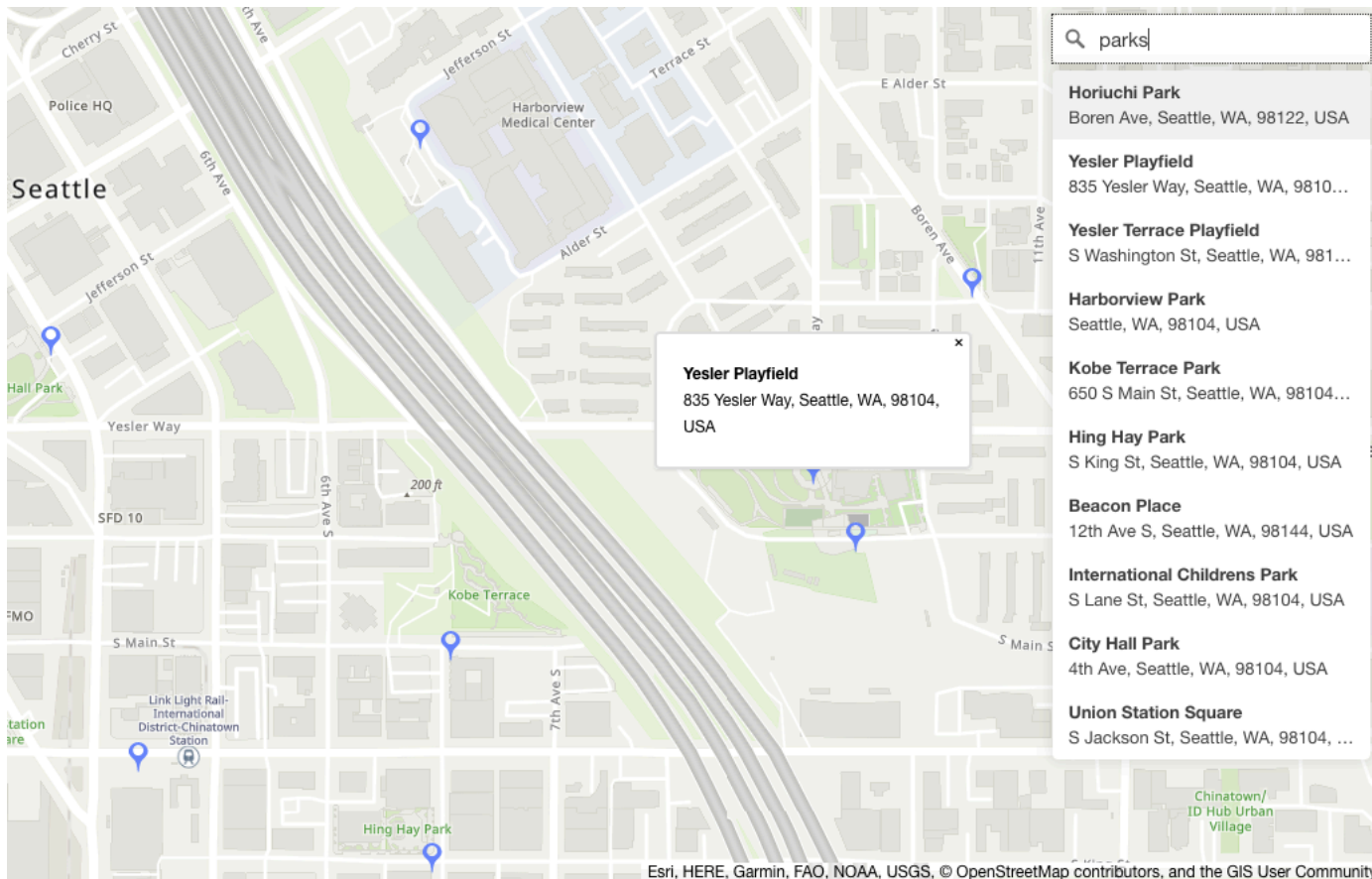
- Utenti IAM:

- Crea un ruolo che l'utente possa assumere. Per istruzioni, consulta la pagina [Creating a role for an IAM user](#) (Creazione di un ruolo per un utente IAM) nella Guida per l'utente di IAM.
- (Non consigliato) Collega una policy direttamente a un utente o aggiungi un utente a un gruppo di utenti. Segui le istruzioni riportate nella pagina [Aggiunta di autorizzazioni a un utente \(console\)](#) nella Guida per l'utente di IAM.

Quando crei applicazioni che utilizzano Amazon Location Service, potrebbe essere necessario che alcuni utenti dispongano di un accesso non autenticato. Per questi casi d'uso, consulta [Abilitare l'accesso non autenticato utilizzando Amazon Cognito](#).

## Utilizzo di Amazon Location Maps nella tua applicazione

Le mappe di localizzazione di Amazon sono convenienti e interattive. Puoi sostituire una mappa esistente nell'applicazione per risparmiare denaro o aggiungerne una nuova per visualizzare visivamente i dati basati sulla posizione, come la posizione del tuo negozio.



Amazon Location Service ti consente di scegliere un fornitore di dati per le operazioni cartografiche creando e configurando una risorsa cartografica. La risorsa cartografica configura il fornitore di dati e lo stile utilizzato per il rendering della mappa.

Dopo aver creato la risorsa, puoi inviare richieste utilizzando direttamente l'AWSSDK o utilizzando una libreria creata appositamente per il rendering delle mappe nel tuo ambiente.

### Note

Per una panoramica dei concetti relativi alle mappe, consulta [Mappe](#).

## Argomenti

- [Prerequisiti](#)
- [Visualizzare una mappa nell'applicazione](#)
- [Disegnare caratteristiche dei dati su una mappa](#)
- [Impostazione delle estensioni per una mappa utilizzando MapLibre](#)
- [Gestione delle risorse cartografiche](#)

## Prerequisiti

Prima di visualizzare una mappa nell'applicazione, segui i passaggi necessari:

### Argomenti

- [Crea una risorsa cartografica](#)
- [Autenticazione delle richieste](#)

## Crea una risorsa cartografica

Per utilizzare una mappa nell'applicazione è necessario disporre di una risorsa cartografica che specifichi lo stile della mappa e il fornitore di dati da utilizzare nelle mappe.

### Note

Se la tua applicazione esegue il tracciamento o l'instradamento di risorse utilizzate nella tua azienda, ad esempio veicoli per la consegna o dipendenti, non devi utilizzare Esri come provider di geolocalizzazione. Per ulteriori dettagli, consulta la sezione 82 dei [Termini di servizio di AWS](#).

Puoi creare una risorsa cartografica utilizzando la console Amazon Location ServiceAWS CLI, o le API di Amazon Location.

### Console

Per creare una risorsa cartografica utilizzando la console Amazon Location Service

1. Nella console Amazon Location, nella pagina [Mappe](#), scegli Crea mappa per visualizzare in anteprima gli stili delle mappe.

2. Aggiungi un nome e una descrizione per la nuova risorsa cartografica.
3. Scegli uno stile di mappa.

**Note**

Se la tua applicazione esegue il tracciamento o l'instradamento di risorse utilizzate nella tua azienda, ad esempio veicoli per la consegna o dipendenti, non devi utilizzare Esri come provider di geolocalizzazione. Per ulteriori dettagli, consulta la sezione 82 dei [Termini di servizio di AWS](#).

4. Scegli tra quelli [Opinioni politiche](#) da usare.
5. Accetta i Termini e condizioni di Amazon Location, quindi scegli Crea mappa. Puoi interagire con la mappa che hai scelto: ingrandisci, rimpicciolisci o fai una panoramica in qualsiasi direzione.
6. Per consentire agli utenti di cambiare stile (ad esempio, per consentire loro di passare dalle immagini satellitari allo stile vettoriale), devi creare una risorsa cartografica per ogni stile.

Puoi eliminare le risorse con stili di mappa che non desideri utilizzare nella [home page di Maps](#) nella console.

## API

Per creare una risorsa cartografica utilizzando le API di Amazon Location

Usa l'[CreateMap](#) operazione dalle API di Amazon Location.

L'esempio seguente è una richiesta API per creare una risorsa cartografica chiamata *ExampleMap* utilizzando lo stile della *VectorEsriStreets* mappa.

```
POST /maps/v0/maps HTTP/1.1
Content-type: application/json

{
  "Configuration": {
    "Style": "VectorEsriStreets"
  },
  "MapName": "ExampleMap"
}
```

**Note**

Se la tua applicazione esegue il tracciamento o l'instradamento di risorse utilizzate nella tua azienda, ad esempio veicoli per la consegna o dipendenti, non devi utilizzare Esri come provider di geolocalizzazione. Per ulteriori dettagli, consulta la sezione 82 dei [Termini di servizio di AWS](#).

## AWS CLI

Per creare una risorsa cartografica utilizzando AWS CLI i comandi

Utilizza il comando [create-map](#).

L'esempio seguente crea una risorsa cartografica chiamata *ExampleMap* using *VectorEsriStreets* come stile della mappa.

```
aws location \  
  create-map \  
  --configuration Style="VectorEsriStreets" \  
  --map-name "ExampleMap"
```

**Note**

Se la tua applicazione esegue il tracciamento o l'instradamento di risorse utilizzate nella tua azienda, ad esempio veicoli per la consegna o dipendenti, non devi utilizzare Esri come provider di geolocalizzazione. Per ulteriori dettagli, consulta la sezione 82 dei [Termini di servizio di AWS](#).

## Autenticazione delle richieste

Una volta creata una risorsa cartografica e sei pronto per iniziare a inserire funzionalità di localizzazione nella tua applicazione, devi scegliere come autenticare le tue richieste.

**Note**

La maggior parte delle applicazioni front-end per mappe richiede l'accesso non autenticato alle mappe o ad altre funzionalità di Amazon Location Service. A seconda dell'applicazione, potresti voler utilizzare AWS Signature v4 per autenticare le richieste oppure puoi utilizzare

le chiavi API Amazon Cognito o Amazon Location per un utilizzo non autenticato. Per ulteriori informazioni su tutte queste opzioni, consulta. [Concessione dell'accesso ad Amazon Location Service](#)

## Visualizzare una mappa nell'applicazione

Questa sezione fornisce tutorial su come utilizzare gli strumenti di rendering delle mappe per visualizzare una mappa nella tua applicazione mobile o web quando usi le API di Amazon Location. Come indicato nell'[Come usare Amazon Location Service](#) argomento, puoi scegliere tra diverse librerie da utilizzare per il rendering di mappe con Amazon Location, tra cui Amplify MapLibre e Tangram.

Effettua una delle seguenti operazioni per visualizzare una mappa nell'applicazione:

- Il modo più diretto per visualizzare una mappa nelle applicazioni front-end Web e mobili è utilizzare MapLibre. Puoi seguire i [MapLibre tutorial](#) o anche il [tutorial di avvio rapido](#) per imparare a usare MapLibre
- Se sei uno AWS Amplify sviluppatore esistente, potresti voler utilizzare l'Amplify Geo SDK. Per saperne di più, segui il tutorial [Amplify](#).
- Se sei già un utente di Tangram e desideri continuare a utilizzarlo per il rendering della tua mappa, mentre passi ad Amazon Location Service, segui il tutorial di [Tangram](#).

### Argomenti

- [Utilizzo della MapLibre libreria con Amazon Location Service](#)
- [Utilizzo della libreria Amplify con Amazon Location Service](#)
- [Usare Tangram con Amazon Location Service](#)

## Utilizzo della MapLibre libreria con Amazon Location Service

I seguenti tutorial illustrano l'utilizzo della MapLibre Library with Amazon Location.

### Argomenti

- [Utilizzo di MapLibre GL JS con Amazon Location Service](#)
- [Utilizzo dell'SDK MapLibre nativo per Android con Amazon Location Service](#)
- [Utilizzo dell'SDK MapLibre nativo per iOS con Amazon Location Service](#)

## Utilizzo di MapLibre GL JS con Amazon Location Service

Usa [MapLibre GL JS](#) per incorporare mappe lato client nelle applicazioni web.

MapLibre GL JS è una JavaScript libreria open source compatibile con gli stili e i riquadri forniti dall'API Maps di Amazon Location Service. Puoi integrare MapLibre GL JS in un codice HTML o in un' JavaScript applicazione di base per incorporare mappe lato client personalizzabili e reattive.

Questo tutorial descrive come integrare MapLibre GL JS con Amazon Location all'interno di un codice HTML e un' JavaScript applicazione di base. [Le stesse librerie e tecniche presentate in questo tutorial si applicano anche ai framework, come React e Angular.](#)

L'applicazione di esempio per questo tutorial è disponibile come parte dell'archivio di esempi di Amazon Location Service su [GitHub](#).

### Creazione dell'applicazione: Scaffolding

Questo tutorial crea un'applicazione web che utilizza JavaScript per creare una mappa su una pagina HTML.

Inizia creando una pagina HTML (`index.html`) che includa il contenitore della mappa:

- Inserisci un `div` elemento con un `id` of `map` per applicare le dimensioni della mappa alla visualizzazione della mappa. Le dimensioni vengono ereditate dalla finestra.

```
<html>
  <head>
    <style>
      body {
        margin: 0;
      }

      #map {
        height: 100vh; /* 100% of viewport height */
      }
    </style>
  </head>
  <body>
    <!-- map container -->
    <div id="map" />
  </body>
```



```
</html>
```

Creazione dell'applicazione: aggiunta di dipendenze

Aggiungi le dipendenze seguenti all'applicazione:

- MapLibre GL JS (v3.x) e il relativo CSS associato.
- La sede di Amazon [JavaScript Aiutante di autenticazione](#).

```
<!-- CSS dependencies -->
<link
  href="https://unpkg.com/maplibre-gl@3.x/dist/maplibre-gl.css"
  rel="stylesheet"
/>
<!-- JavaScript dependencies -->
<script src="https://unpkg.com/maplibre-gl@3.x/dist/maplibre-gl.js"></script>
<script src="https://unpkg.com/@aws/amazon-location-authentication-helper.js"></script>
<script>
  // application-specific code
</script>
```

Questo crea una pagina vuota con il contenitore della mappa.

Creazione dell'applicazione: configurazione

Per configurare l'applicazione utilizzando JavaScript:

1. Inserisci i nomi e gli identificatori delle tue risorse.

```
// Cognito Identity Pool ID
const identityPoolId = "us-east-1:54f2ba88-9390-498d-aaa5-0d97fb7ca3bd";
// Amazon Location Service Map name
const mapName = "ExampleMap";
```

2. Crea un'istanza di un provider di credenziali utilizzando il pool di identità non autenticato che hai creato in [Uso delle mappe - Passaggio 2](#), Configurazione dell'autenticazione. Lo inseriremo in una funzione chiamata `initializeMap`, che conterrà anche altro codice di inizializzazione della mappa, aggiunto nel passaggio successivo

```
// extract the Region from the Identity Pool ID; this will be used for both Amazon
Cognito and Amazon Location
```

```
AWS.config.region = identityPoolId.split(":")[0];

async function initializeMap() {
  // Create an authentication helper instance using credentials from Cognito
  const authHelper = await
  amazonLocationAuthHelper.withIdentityPoolId(identityPoolId);

  // ... more here, later
}
```

## Creazione dell'applicazione: inizializzazione della mappa

Affinché la mappa venga visualizzata dopo il caricamento della pagina, è necessario inizializzare la mappa. È possibile modificare la posizione iniziale della mappa, aggiungere controlli aggiuntivi e sovrapporre i dati.

```
async function initializeMap() {
  // Create an authentication helper instance using credentials from Cognito
  const authHelper = await amazonLocationAuthHelper.withIdentityPoolId(identityPoolId);

  // Initialize the map
  const map = new maplibregl.Map({
    container: "map",
    center: [-123.1187, 49.2819], // initial map centerpoint
    zoom: 10, // initial map zoom
    style: 'https://maps.geo.${region}.amazonaws.com/maps/v0/maps/${mapName}/style-
descriptor',
    ...authHelper.getMapAuthenticationOptions(), // authentication, using cognito
  });

  map.addControl(new maplibregl.NavigationControl(), "top-left");
}

initializeMap();
```

### Note

È necessario fornire un marchio denominativo o un'attribuzione di testo per ogni fornitore di dati che si utilizza, nell'applicazione o nella documentazione. Le stringhe di attribuzione sono incluse nella risposta del descrittore di stile sotto i tasti `sources.esri.attribution`, `sources.here.attribution` e `sources.grabmaptiles.attribution` MapLibre GL

JS fornirà automaticamente l'attribuzione. Quando utilizzi le risorse di Amazon Location con [fornitori di dati](#), assicurati di leggere i [termini e le condizioni del servizio](#).

## Esecuzione dell'applicazione

È possibile eseguire questa applicazione di esempio utilizzandola in un server Web locale o aprendola in un browser.

Per utilizzare un server web locale, puoi usare npx, perché è installato come parte di Node.js. È possibile utilizzare npx serve dall'interno della stessa directory di `index.html`. Questo serve all'applicazione su `localhost:5000`.

### Note

Se la policy che hai creato per il tuo ruolo non autenticato in Amazon Cognito include `referer` una condizione, potresti essere bloccato dal test con gli URL `localhost`: In questo caso, puoi eseguire il test con un server Web che fornisce un URL indicato nella tua politica.

Dopo aver completato il tutorial, l'applicazione finale è simile all'esempio seguente.

```
<!-- index.html -->
<html>
  <head>
    <link href="https://unpkg.com/maplibre-gl@3.x/dist/maplibre-gl.css"
rel="stylesheet" />
    <style>
      body {
        margin: 0;
      }
      #map {
        height: 100vh;
      }
    </style>
  </head>

  <body>
    <!-- map container -->
```

```
<div id="map" />
<!-- JavaScript dependencies -->
<script src="https://unpkg.com/maplibre-gl@3.x/dist/maplibre-gl.js"></script>
<script src="https://unpkg.com/@aws/amazon-location-authentication-helper.js"></
script>
<script>
  // configuration
  const identityPoolId = "us-east-1:54f2ba88-9390-498d-aaa5-0d97fb7ca3bd"; //
Cognito Identity Pool ID
  const mapName = "ExampleMap"; // Amazon Location Service Map Name

  // extract the region from the Identity Pool ID
  const region = identityPoolId.split(":")[0];

  async function initializeMap() {
    // Create an authentication helper instance using credentials from Cognito
    const authHelper = await
amazonLocationAuthHelper.withIdentityPoolId(identityPoolId);

    // Initialize the map
    const map = new maplibregl.Map({
      container: "map",
      center: [-123.115898, 49.295868],
      zoom: 10,
      style: `https://maps.geo.${region}.amazonaws.com/maps/v0/maps/${mapName}/
style-descriptor`,
      ...authHelper.getMapAuthenticationOptions(),
    });
    map.addControl(new maplibregl.NavigationControl(), "top-left");
  }

  initializeMap();
</script>
</body>
</html>
```

L'esecuzione di questa applicazione consente di visualizzare una mappa a schermo intero utilizzando lo stile di mappa scelto. Questo esempio è disponibile nell'archivio degli esempi di Amazon Location Service su [GitHub](#).

Utilizzo dell'SDK MapLibre nativo per Android con Amazon Location Service

Usa [MapLibreNative](#) SDK per incorporare mappe interattive nelle tue applicazioni Android.

L'SDK MapLibre nativo per Android è una libreria basata [su Mapbox Native](#) ed è compatibile con gli stili e i riquadri forniti dall'API Maps di Amazon Location Service. Puoi integrare MapLibre Native SDK for Android per incorporare visualizzazioni di mappe interattive con mappe vettoriali scalabili e personalizzabili nelle tue applicazioni Android.

Questo tutorial descrive come integrare l'SDK MapLibre nativo per Android con Amazon Location. L'applicazione di esempio per questo tutorial è disponibile come parte dell'archivio di esempi di Amazon Location Service su [GitHub](#).

Creazione dell'applicazione: inizializzazione

Per inizializzare l'applicazione:

1. Crea un nuovo progetto Android Studio dal modello Empty Activity.
2. Assicurati che Kotlin sia selezionato per la lingua del progetto.
3. Seleziona un SDK minimo di API 14: Android 4.0 (Ice Cream Sandwich) o versione successiva.
4. Apri Project Structure, quindi vai su File > Struttura del progetto... per scegliere la sezione Dipendenze.
5. Con <All Modules>selezionato, quindi scegli il pulsante + per aggiungere una nuova dipendenza dalla libreria.
6. Aggiungi AWS Android SDK versione 2.20.0 o successiva. Ad esempio: `com.amazonaws:aws-android-sdk-core:2.20.0`
7. Aggiungi l'SDK MapLibre nativo per Android versione 9.4.0 o successiva. Ad esempio: `org.maplibre.gl:android-sdk:9.4.0`
8. A livello di progetto del file `build.gradle`, aggiungi il seguente repository maven per accedere ai pacchetti per Android: MapLibre

```
allprojects {
    repositories {
        // Retain your existing repositories
        google()
        jcenter()

        // Declare the repositories for MapLibre
        mavenCentral()
    }
}
```

## Creazione dell'applicazione: configurazione

Per configurare l'applicazione con le tue risorse e la tua AWS regione:

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
  <string name="identityPoolId">us-east-1:54f2ba88-9390-498d-aaa5-0d97fb7ca3bd</string>
  <string name="mapName">ExampleMap</string>
  <string name="awsRegion">us-east-1</string>
</resources>
```

## Creazione dell'applicazione: Activity layout

Modifica `app/src/main/res/layout/activity_main.xml`:

- Aggiungi un `MapView`, che rende la mappa. Questo imposterà anche il punto centrale iniziale della mappa.
- Aggiungi un `TextView`, che mostra l'attribuzione.

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
  xmlns:android="http://schemas.android.com/apk/res/android"
  xmlns:app="http://schemas.android.com/apk/res-auto"
  xmlns:tools="http://schemas.android.com/tools"
  android:layout_width="match_parent"
  android:layout_height="match_parent"
  tools:context=".MainActivity">

  <com.mapbox.mapboxsdk.maps.MapView
    android:id="@+id/mapView"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    app:mapbox_cameraTargetLat="49.2819"
    app:mapbox_cameraTargetLng="-123.1187"
    app:mapbox_cameraZoom="12"
    app:mapbox_uiAttribution="false"
    app:mapbox_uiLogo="false" />

  <TextView
    android:id="@+id/attributionView"
    android:layout_width="wrap_content"
```

```
    android:layout_height="wrap_content"
    android:background="#80808080"
    android:padding="5sp"
    android:textColor="@android:color/black"
    android:textSize="10sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    tools:ignore="SmallSp" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

### Note

È necessario fornire un marchio denominativo o un'attribuzione di testo per ogni fornitore di dati che si utilizza, nell'applicazione o nella documentazione. Le stringhe di attribuzione sono incluse nella risposta del descrittore di stile sotto i tasti `sources.esri.attribution`, `sources.here.attribution` e `source.grabmaptiles.attribution`. Quando utilizzi le risorse di Amazon Location con [fornitori di dati](#), assicurati di leggere i [termini e le condizioni del servizio](#).

Creazione dell'applicazione: richiedi la trasformazione

Crea una classe denominata `SigV4Interceptor` per intercettare le richieste AWS e firmala utilizzando [Signature Version 4](#). Questa verrà registrata con il client HTTP utilizzato per recuperare le risorse della mappa quando viene creata l'attività principale.

```
package aws.location.demo.okhttp

import com.amazonaws.DefaultRequest
import com.amazonaws.auth.AWS4Signer
import com.amazonaws.auth.AWSCredentialsProvider
import com.amazonaws.http.HttpMethodName
import com.amazonaws.util.IOUtils
import okhttp3.HttpUrl
import okhttp3.Interceptor
import okhttp3.Request
import okhttp3.Response
import okio.Buffer
import java.io.ByteArrayInputStream
import java.net.URI
```

```
class SigV4Interceptor(
    private val credentialsProvider: AWSCredentialsProvider,
    private val serviceName: String
) : Interceptor {
    override fun intercept(chain: Interceptor.Chain): Response {
        val originalRequest = chain.request()

        if (originalRequest.url().host().contains("amazonaws.com")) {
            val signer = if (originalRequest.url().encodedPath().contains("@")) {
                // the presence of "@" indicates that it doesn't need to be double URL-
                encoded
                AWS4Signer(false)
            } else {
                AWS4Signer()
            }

            val awsRequest = toAWSRequest(originalRequest, serviceName)
            signer.setServiceName(serviceName)
            signer.sign(awsRequest, credentialsProvider.credentials)

            return chain.proceed(toSignedOkHttpRequest(awsRequest, originalRequest))
        }

        return chain.proceed(originalRequest)
    }

    companion object {
        fun toAWSRequest(request: Request, serviceName: String): DefaultRequest<Any> {
            // clone the request (AWS-style) so that it can be populated with
            credentials
            val dr = DefaultRequest<Any>(serviceName)

            // copy request info
            dr.httpMethod = HttpMethodName.valueOf(request.method())
            with(request.url()) {
                dr.resourcePath = uri().path
                dr.endpoint = URI.create("${scheme()}://${host()}")

                // copy parameters
                for (p in queryParameterNames()) {
                    if (p != "") {
                        dr.addParameter(p, queryParameter(p))
                    }
                }
            }
        }
    }
}
```



```
    }

    // copy headers
    for (h in request.headers().names()) {
        dr.addHeader(h, request.header(h))
    }

    // copy the request body
    val bodyBytes = request.body()?.let { body ->
        val buffer = Buffer()
        body.writeTo(buffer)
        IOUtils.toByteArray(buffer.inputStream())
    }

    dr.content = ByteArrayInputStream(bodyBytes ?: ByteArray(0))

    return dr
}

fun toSignedOkHttpRequest(
    awsRequest: DefaultRequest<Any>,
    originalRequest: Request
): Request {
    // copy signed request back into an OkHttp Request
    val builder = Request.Builder()

    // copy headers from the signed request
    for ((k, v) in awsRequest.headers) {
        builder.addHeader(k, v)
    }

    // start building an HttpUrl
    val urlBuilder = HttpUrl.Builder()
        .host(awsRequest.endpoint.host)
        .scheme(awsRequest.endpoint.scheme)
        .encodedPath(awsRequest.resourcePath)

    // copy parameters from the signed request
    for ((k, v) in awsRequest.parameters) {
        urlBuilder.addQueryParameter(k, v)
    }

    return builder.url(urlBuilder.build())
        .method(originalRequest.method(), originalRequest.body())
}
```

```
        .build()
    }
}
```

## Creazione dell'applicazione: attività principale

L'attività principale è responsabile dell'inizializzazione delle visualizzazioni che verranno mostrate agli utenti. Ciò comporta:

- Creazione di un'istanza di Amazon Cognito. `CredentialsProvider`
- Registrazione dell'interceptor `Signature Version 4`.
- Configurazione della mappa puntandola verso un descrittore di stile della mappa e visualizzando l'attribuzione appropriata.

`MainActivity` è anche responsabile dell'inoltro degli eventi del ciclo di vita alla visualizzazione della mappa, consentendole di preservare la finestra attiva tra le chiamate.

```
package aws.location.demo.maplibre

import android.os.Bundle
import android.widget.TextView
import androidx.appcompat.app.AppCompatActivity
import aws.location.demo.okhttp.SigV4Interceptor
import com.amazonaws.auth.CognitoCachingCredentialsProvider
import com.amazonaws.regions.Regions
import com.mapbox.mapboxsdk.Mapbox
import com.mapbox.mapboxsdk.maps.MapView
import com.mapbox.mapboxsdk.maps.Style
import com.mapbox.mapboxsdk.module.http.HttpRequestUtil
import okhttp3.OkHttpClient

private const val SERVICE_NAME = "geo"

class MainActivity : AppCompatActivity() {
    private var mapView: MapView? = null

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)

        // configuration
```

```
val identityPoolId = getString(R.string.identityPoolId)
val region = getString(R.string.awsRegion)
val mapName = getString(R.string.mapName)

// Credential initialization
val credentialProvider = CognitoCachingCredentialsProvider(
    applicationContext,
    identityPoolId,
    Regions.fromName(identityPoolId.split(":").first())
)

// initialize MapLibre
Mapbox.getInstance(this, null)
HttpRequestUtil.setOkHttpClient(
    OkHttpClient.Builder()
        .addInterceptor(SigV4Interceptor(credentialProvider, SERVICE_NAME))
        .build()
)

// initialize the view
setContentView(R.layout.activity_main)

// initialize the map view
mapView = findViewById(R.id.mapView)
mapView?.onCreate(savedInstanceState)
mapView?.getMapAsync { map ->
    map.setStyle(
        Style.Builder()
            .fromUri("https://maps.geo.{$region}.amazonaws.com/maps/v0/maps/
{$mapName}/style-descriptor")
    ) { style ->
        findViewById<TextView>(R.id.attributionView).text =
style.sources.first()?.attribution
    }
}

override fun onStart() {
    super.onStart()
    mapView?.onStart()
}

override fun onResume() {
    super.onResume()
}
```

```
        mapView?.onResume()
    }

    override fun onPause() {
        super.onPause()
        mapView?.onPause()
    }

    override fun onStop() {
        super.onStop()
        mapView?.onStop()
    }

    override fun onSaveInstanceState(outState: Bundle) {
        super.onSaveInstanceState(outState)
        mapView?.onSaveInstanceState(outState)
    }

    override fun onLowMemory() {
        super.onLowMemory()
        mapView?.onLowMemory()
    }

    override fun onDestroy() {
        super.onDestroy()
        mapView?.onDestroy()
    }
}
```

L'esecuzione di questa applicazione consente di visualizzare una mappa a schermo intero nello stile desiderato. Questo esempio è disponibile come parte dell'archivio di esempi di Amazon Location Service su [GitHub](#).

### Utilizzo dell'SDK MapLibre nativo per iOS con Amazon Location Service

Usa [MapLibreNative SDK for iOS per](#) incorporare mappe lato client nelle applicazioni iOS.

L'SDK MapLibre nativo per iOS è una libreria basata [su Mapbox GL](#) Native ed è compatibile con gli stili e i riquadri forniti dall'API Maps di Amazon Location Service. Puoi integrare MapLibre Native SDK for iOS per incorporare visualizzazioni di mappe interattive con mappe vettoriali scalabili e personalizzabili nelle tue applicazioni iOS.

Questo tutorial descrive come integrare l'SDK MapLibre nativo per iOS con Amazon Location. L'applicazione di esempio per questo tutorial è disponibile come parte dell'archivio di esempi di Amazon Location Service su [GitHub](#).

Creazione dell'applicazione: inizializzazione


Per inizializzare l'applicazione:

1. Crea un nuovo progetto Xcode dal modello dell'app.
2. Seleziona SwiftUI per la sua interfaccia.
3. Seleziona l'applicazione SwiftUI per il suo ciclo di vita.
4. Seleziona Swift per la sua lingua.

Aggiungere MapLibre dipendenze utilizzando i pacchetti Swift

Per aggiungere una dipendenza dal pacchetto al tuo progetto Xcode:

1. Passa a File > Pacchetti Swift > Aggiungi dipendenza dal pacchetto.
2. Inserisci l'URL del repository: **<https://github.com/maplibre/maplibre-gl-native-distribution>**

 Note

Per ulteriori informazioni sui pacchetti Swift, consulta [Adding Package Dependencies to Your App](#) su Apple.com

3. Nel tuo terminale, installa: CocoaPods

```
sudo gem install cocoapods
```

4. Vai alla directory del progetto dell'applicazione e inizializza il Podfile con il gestore di CocoaPods pacchetti:

```
pod init
```

5. Apri il Podfile da aggiungere AWSCore come dipendenza:

```
platform :ios, '12.0'
```

```
target 'Amazon Location Service Demo' do
  use_frameworks!

  pod 'AWSCore'
end
```

## 6. Scarica e installa le dipendenze:

```
pod install --repo-update
```

## 7. Apri lo spazio di lavoro Xcode che ha creato: CocoaPods

```
xed .
```

### Creazione dell'applicazione: configurazione

Aggiungi le seguenti chiavi e valori a Info.plist per configurare l'applicazione:

Chiave	Valore
AWSRegion	us-east-1
IdentityPoolId	us-east- 1:54 f2ba88-9390-498d-aaa5-0d97f b7ca3bd
MapName	ExampleMap

### Creazione dell'applicazione: layout ContentView

Per renderizzare la mappa, modifica `ContentView.swift`:

- Aggiungi un file `MapView` che renderizza la mappa.
- Aggiungi un `TextField` che mostra l'attribuzione.

Questo imposta anche il punto centrale iniziale della mappa.

```
import SwiftUI

struct ContentView: View {
```

```
@State private var attribution = ""

var body: some View {
    MapView(attribution: $attribution)
        .centerCoordinate(.init(latitude: 49.2819, longitude: -123.1187))
        .zoomLevel(12)
        .edgesIgnoringSafeArea(.all)
        .overlay(
            TextField("", text: $attribution)
                .disabled(true)
                .font(.system(size: 12, weight: .light, design: .default))
                .foregroundColor(.black)
                .background(Color.init(Color.RGBColorSpace.sRGB, white: 0.5,
opacity: 0.5))
                .cornerRadius(1),
            alignment: .bottomTrailing)
        }
}

struct ContentView_Previews: PreviewProvider {
    static var previews: some View {
        ContentView()
    }
}
```

### Note

È necessario fornire un marchio denominativo o un'attribuzione di testo per ogni fornitore di dati che si utilizza, nell'applicazione o nella documentazione. Le stringhe di attribuzione sono incluse nella risposta del descrittore di stile sotto i tasti `sources.esri.attribution`, `sources.here.attribution` e `source.grabmaptiles.attribution`. Quando utilizzi le risorse di Amazon Location con [fornitori di dati](#), assicurati di leggere i [termini e le condizioni del servizio](#).

Creazione dell'applicazione: richiedi la trasformazione

Crea un nuovo file Swift denominato `AWSSignatureV4Delegate.swift` contenente la seguente definizione di classe per intercettare le richieste AWS e firmarle [utilizzando Signature Version 4](#). Un'istanza di questa classe verrà assegnata come delegata allo storage offline, che è anche responsabile della riscrittura degli URL, nella visualizzazione della mappa.

```

import AWSCore
import Mapbox

class AWSSignatureV4Delegate : NSObject, MGLOfflineStorageDelegate {
    private let region: AWSRegionType
    private let identityPoolId: String
    private let credentialsProvider: AWSCredentialsProvider

    init(region: AWSRegionType, identityPoolId: String) {
        self.region = region
        self.identityPoolId = identityPoolId
        self.credentialsProvider = AWSCognitoCredentialsProvider(regionType: region,
identityPoolId: identityPoolId)
        super.init()
    }

    class func doubleEncode(path: String) -> String? {
        return path.addingPercentEncoding(withAllowedCharacters: .urlPathAllowed)?
            .addingPercentEncoding(withAllowedCharacters: .urlPathAllowed)
    }

    func offlineStorage(_ storage: MGLOfflineStorage, urlForResourceOf kind:
MGLResourceKind, with url: URL) -> URL {
        if url.host?.contains("amazonaws.com") != true {
            // not an AWS URL
            return url
        }

        // URL-encode spaces, etc.
        let keyPath = String(url.path.dropFirst())
        guard let percentEncodedKeyPath =
keyPath.addingPercentEncoding(withAllowedCharacters: .urlPathAllowed) else {
            print("Invalid characters in path '\(keyPath)'; unsafe to sign")
            return url
        }

        let endpoint = AWSEndpoint(region: region, serviceName: "geo", url: url)
        let requestHeaders: [String: String] = ["host": endpoint!.hostName]

        // sign the URL
        let task = AWSSignatureV4Signer
            .generateQueryStringForSignatureV4(
                withCredentialProvider: credentialsProvider,

```



```

        httpMethod: .GET,
        expireDuration: 60,
        endpoint: endpoint!,
        // workaround for https://github.com/aws-amplify/aws-sdk-ios/
issues/3215
        keyPath: AWSSignatureV4Delegate.doubleEncode(path:
percentEncodedKeyPath),
        requestHeaders: requestHeaders,
        requestParameters: .none,
        signBody: true)
    task.waitUntilFinished()

    if let error = task.error as NSError? {
        print("Error occurred: \(error)")
    }

    if let result = task.result {
        var urlComponents = URLComponents(url: (result as URL),
resolvingAgainstBaseURL: false)!
        // re-use the original path; workaround for https://github.com/aws-amplify/
aws-sdk-ios/issues/3215
        urlComponents.path = url.path

        // have Mapbox GL fetch the signed URL
        return (urlComponents.url)!
    }

    // fall back to an unsigned URL
    return url
}
}

```

## Creazione dell'applicazione: visualizzazione della mappa

Map View è responsabile dell'inizializzazione `AWSSignatureV4Delegate` e della configurazione di un'istanza di quella sottostante `MGLMapView`, che recupera le risorse e rende la mappa. Gestisce inoltre la propagazione delle stringhe di attribuzione dalla fonte del descrittore di stile al `ContentView`

Crea un nuovo file Swift denominato `MapView.swift` contenente la seguente definizione: `struct`

```

import SwiftUI
import AWSCore

```

```

import Mapbox

struct MapView: UIViewRepresentable {
    @Binding var attribution: String

    private var mapView: MGLMapView
    private var signingDelegate: MGLOfflineStorageDelegate

    init(attribution: Binding<String>) {
        let regionName = Bundle.main.object(forKey: "AWSRegion") as!
String
        let identityPoolId = Bundle.main.object(forKey: "IdentityPoolId")
as! String
        let mapName = Bundle.main.object(forKey: "MapName") as! String

        let region = (regionName as NSString).aws_regionTypeValue()

        // MGLOfflineStorage doesn't take ownership, so this needs to be a member here
        signingDelegate = AWSSignatureV4Delegate(region: region, identityPoolId:
identityPoolId)

        // register a delegate that will handle SigV4 signing
        MGLOfflineStorage.shared.delegate = signingDelegate

        mapView = MGLMapView(
            frame: .zero,
            styleURL: URL(string: "https://maps.geo.\(regionName).amazonaws.com/maps/
v0/maps/\(mapName)/style-descriptor"))

        _attribution = attribution
    }

    func makeCoordinator() -> Coordinator {
        Coordinator($attribution)
    }

    class Coordinator: NSObject, MGLMapViewDelegate {
        var attribution: Binding<String>

        init(_ attribution: Binding<String>) {
            self.attribution = attribution
        }

        func mapView(_ mapView: MGLMapView, didFinishLoading style: MGLStyle) {

```

```

        let source = style.sources.first as? MGLVectorTileSource
        let attribution = source?.attributionInfos.first
        self.attribution.wrappedValue = attribution?.title.string ?? ""
    }
}

// MARK: - UIViewRepresentable protocol

func makeUIView(context: UIViewRepresentableContext<MapView>) -> MGLMapView {
    mapView.delegate = context.coordinator

    mapView.logoView.isHidden = true
    mapView.attributionButton.isHidden = true
    return mapView
}

func updateUIView(_ uiView: MGLMapView, context:
UIViewRepresentableContext<MapView>) {
}

// MARK: - MGLMapView proxy

func centerCoordinate(_ centerCoordinate: CLLocationCoordinate2D) -> MapView {
    mapView.centerCoordinate = centerCoordinate
    return self
}

func zoomLevel(_ zoomLevel: Double) -> MapView {
    mapView.zoomLevel = zoomLevel
    return self
}
}

```

L'esecuzione di questa applicazione mostra una mappa a schermo intero nello stile che preferisci. Questo esempio è disponibile come parte dell'archivio di esempi di Amazon Location Service su [GitHub](#).

## Utilizzo della libreria Amplify con Amazon Location Service

Il seguente tutorial illustra l'utilizzo AWS Amplify con Amazon Location. Amplify MapLibre utilizza GL JS per renderizzare le mappe nella tua applicazione basata. JavaScript

Amplify è un insieme di librerie client open source che forniscono interfacce a diverse categorie di servizi, tra cui Amplify Geo, che è alimentato da Amazon Location Service. [Scopri di più sulla JavaScript libreria AWS Amplify Geo.](#)

### Note

Questo tutorial presuppone che tu abbia già seguito i passaggi descritti in [Uso delle mappe - Per aggiungere una mappa all'applicazione.](#)

## Creazione dell'applicazione: impalcature

Questo tutorial crea un'applicazione web che utilizza JavaScript per creare una mappa su una pagina HTML.

Inizia creando una pagina HTML (`index.html`) che includa il contenitore della mappa:

- Inserisci un `div` elemento con un `id` of `map` per applicare le dimensioni della mappa alla visualizzazione della mappa. Le dimensioni vengono ereditate dalla finestra.

```
<html>
  <head>
    <style>
      body { margin: 0; }
      #map { height: 100vh; } /* 100% of viewport height */
    </style>
  </head>

  <body>
    <!-- map container -->
    <div id="map" />
  </body>
</html>
```

## Creazione dell'applicazione: aggiunta di dipendenze

Aggiungi le dipendenze seguenti all'applicazione:

- AWS Amplify librerie di mappe e geografiche.
- AWS Amplify libreria principale.

- AWS Amplifylibreria di autenticazione.
- AWS Amplifyfoglio di stile.

```
<!-- CSS dependencies -->
  <link href="https://cdn.amplify.aws/packages/maplibre-
gl/1.15.2/maplibre-gl.css" rel="stylesheet" integrity="sha384-
DrPVD9GufrxGb7kWwRv0CywpXTmfvbK0Z5i5pN7urmIThew0zXKTME+gutUgtpeD"
  crossorigin="anonymous" referrerpolicy="no-referrer"></link>

<!-- JavaScript dependencies -->
  <script src="https://cdn.amplify.aws/packages/maplibre-gl/1.15.2/maplibre-gl.js"
  integrity="sha384-rwYfkmA0pciZS2bDuwZ/Xa/Gog6jXem8D/whm3wnsZSVFemDDLprcUXHnDDUcrNU"
  crossorigin="anonymous" referrerpolicy="no-referrer"></script>
  <script src="https://cdn.amplify.aws/packages/core/4.3.0/aws-amplify-core.min.js"
  integrity="sha384-70h+5w0l7XGyYvSqbkKi2Q7SA5K640V5nyW2/LEbevDQEV1HMQJLA1A00z2hu8fJ"
  crossorigin="anonymous" referrerpolicy="no-referrer"></script>
  <script src="https://cdn.amplify.aws/packages/auth/4.3.8/aws-amplify-auth.min.js"
  integrity="sha384-jfkXCEfYyVmDXyKlGWNwv54xRaZgk14m7sjeb2jLVBtUXCD2p+WU8YZ2mPZ9Xbdw"
  crossorigin="anonymous" referrerpolicy="no-referrer"></script>
  <script src="https://cdn.amplify.aws/packages/geo/1.1.0/aws-amplify-geo.min.js"
  integrity="sha384-TFMtyWuCbiptXTzv0gzJbV8TPUpG1rA1AVrznAhCSpXTIdGw82bGd8RTk5rr3nP"
  crossorigin="anonymous" referrerpolicy="no-referrer"></script>
  <script src="https://cdn.amplify.aws/packages/maplibre-gl-js-
amplify/1.1.0/maplibre-gl-js-amplify.umd.min.js" integrity="sha384-7/
RxWonKW1nM9zCKiwU9x6bkQTjldosg0D1vZYm0Zj+K/vUSnA3s0MhlRRWAtHPi" crossorigin="anonymous"
  referrerpolicy="no-referrer"></script>
<script>
  // application-specific code
</script>
```

Questo crea una pagina vuota con il contenitore della mappa.

Creazione dell'applicazione: configurazione

Per configurare l'applicazione utilizzando JavaScript:

1. Inserisci gli identificatori del pool di identità non autenticato che hai creato in [Utilizzo delle mappe - Passaggio 2, Configurazione](#) dell'autenticazione.

```
// Cognito Identity Pool ID
const identityPoolId = "region:identityPoolID"; // for example: us-
east-1:123example-1234-5678
```

```
// extract the Region from the Identity Pool ID
const region = identityPoolId.split(":")[0];
```

2. Configura AWS Amplify per utilizzare le risorse che hai creato, incluso il pool di identità e la risorsa Map (mostrata qui con il nome predefinito di). `explore.map`

```
// Configure Amplify
const { Amplify } = aws_amplify_core;
const { createMap } = AmplifyMapLibre;

Amplify.configure({
  Auth: {
    identityPoolId,
    region,
  },
  geo: {
    AmazonLocationService: {
      maps: {
        items: {
          "explore.map": {
            style: "Default style"
          },
        },
        default: "explore.map",
      },
      region,
    },
  }
});
```

### Creazione dell'applicazione: inizializzazione della mappa

Affinché la mappa venga visualizzata dopo il caricamento della pagina, è necessario inizializzare la mappa. È possibile modificare la posizione iniziale della mappa, aggiungere controlli aggiuntivi e sovrapporre i dati.

```
async function initializeMap() {
  const map = await createMap(
    {
      container: "map",
      center: [-123.1187, 49.2819],
      zoom: 10,
```

```
        hash: true,  
    }  
);  
  
    map.addControl(new maplibregl.NavigationControl(), "top-left");  
}  
  
initializeMap();
```

### Note

È necessario fornire un marchio denominativo o un'attribuzione di testo per ogni fornitore di dati che si utilizza, nell'applicazione o nella documentazione. Le stringhe di attribuzione sono incluse nella risposta del descrittore di stile sotto i tasti `sources.esri.attribution`, `sources.here.attribution` e `sources.grabmaptiles.attribution`. Amplify fornirà automaticamente l'attribuzione. Quando utilizzi le risorse di Amazon Location con [fornitori di dati](#), assicurati di leggere i [termini e le condizioni del servizio](#).

## Esecuzione dell'applicazione

È possibile eseguire questa applicazione di esempio utilizzandola in un server Web locale o aprendola in un browser.

Per utilizzare un server web locale, puoi usare `npx`, installato come parte di Node.js, o qualsiasi altro server web a tua scelta. Per usare `npx`, digita `npx serve` dalla stessa directory di `index.html`. Questo serve all'applicazione su `localhost:5000`.

### Note

Se la policy che hai creato per il tuo ruolo non autenticato in Amazon Cognito include `referer` una condizione, potresti essere bloccato dal test con gli URL `localhost`: In questo caso, puoi eseguire il test con un server Web che fornisce un URL indicato nella tua politica.

Dopo aver completato il tutorial, l'applicazione finale è simile all'esempio seguente.

```
<html>  
  <head>
```

```

<!-- CSS dependencies -->
<link href="https://cdn.amplify.aws/packages/maplibre-
gl/1.15.2/maplibre-gl.css" rel="stylesheet" integrity="sha384-
DrPVD9GufrxGb7kWwRv0CywXTmfvbK0Z5i5pN7urmIThew0zXKTME+gutUgtpeD"
crossorigin="anonymous" referrerpolicy="no-referrer"></link>

<!-- JavaScript dependencies -->
<script src="https://cdn.amplify.aws/packages/maplibre-gl/1.15.2/maplibre-gl.js"
integrity="sha384-rwYfkmA0pciZS2bDuwZ/Xa/Gog6jXem8D/whm3wnsZSVFemDDlprcUXHnDDUcrNU"
crossorigin="anonymous" referrerpolicy="no-referrer"></script>
<script src="https://cdn.amplify.aws/packages/core/4.3.0/aws-amplify-core.min.js"
integrity="sha384-70h+5w0l7XGyYvSqBki2Q7SA5K640V5nyW2/LEbevDQEV1HMJqJLA1A00z2hu8fJ"
crossorigin="anonymous" referrerpolicy="no-referrer"></script>
<script src="https://cdn.amplify.aws/packages/auth/4.3.8/aws-amplify-auth.min.js"
integrity="sha384-jfkXCEfYyVmDXyKlgWNwv54xRaZgk14m7sjeb2jLVBtUXCD2p+WU8YZ2mPZ9Xbdw"
crossorigin="anonymous" referrerpolicy="no-referrer"></script>
<script src="https://cdn.amplify.aws/packages/geo/1.1.0/aws-amplify-geo.min.js"
integrity="sha384-TFMTyWuCbiptXTzv0gzJbV8TPUupG1rA1AVrznAhCSpXTIdGw82bGd8RTk5rr3nP"
crossorigin="anonymous" referrerpolicy="no-referrer"></script>
<script src="https://cdn.amplify.aws/packages/maplibre-gl-js-
amplify/1.1.0/maplibre-gl-js-amplify.umd.min.js" integrity="sha384-7/
RxWonKW1nM9zCKiwU9x6bkQTjldosg0D1vZYm0Zj+K/vUSnA3s0MhlRRWAtHPi" crossorigin="anonymous"
referrerpolicy="no-referrer"></script>

<style>
  body { margin: 0; }
  #map { height: 100vh; }
</style>
</head>

<body>
  <div id="map" />
  <script type="module">
    // Cognito Identity Pool ID
    const identityPoolId = "region:identityPoolId"; // for example: us-
east-1:123example-1234-5678
    // extract the Region from the Identity Pool ID
    const region = identityPoolId.split(":")[0];

    // Configure Amplify
    const { Amplify } = aws_amplify_core;
    const { createMap } = AmplifyMapLibre;

    Amplify.configure({

```



```
Auth: {
  identityPoolId,
  region,
},
geo: {
  AmazonLocationService: {
    maps: {
      items: {
        "explore.map": {
          style: "Default style"
        },
      },
      default: "explore.map",
    },
    region,
  },
}
});

async function initializeMap() {
  const map = await createMap(
    {
      container: "map",
      center: [-123.1187, 49.2819],
      zoom: 10,
      hash: true,
    }
  );

  map.addControl(new maplibregl.NavigationControl(), "top-left");
}

initializeMap();
</script>
</body>
</html>
```

L'esecuzione di questa applicazione consente di visualizzare una mappa a schermo intero utilizzando lo stile di mappa scelto. Questo esempio è descritto anche nella scheda [Incorpora mappa di qualsiasi pagina di risorse della mappa nella console Amazon Location Service](#).

Dopo aver completato questo tutorial, vai all'argomento [Visualizza una mappa](#) nella AWS Amplify documentazione per saperne di più, incluso come visualizzare gli indicatori sulla mappa.

## Usare Tangram con Amazon Location Service

Questa sezione fornisce i seguenti tutorial su come integrare Tangram con Amazon Location.

### ⚠ Important

Gli stili Tangram nei seguenti tutorial sono compatibili solo con le risorse della mappa di Amazon Location configurate con lo stile. `VectorHereContrast`

Di seguito è riportato un esempio di comando AWS CLI per creare una nuova risorsa cartografica chiamata *TangramExampleMap* utilizzando lo *VectorHereContrast* stile:

```
aws --region us-east-1 \  
  location \  
  create-map \  
  --map-name "TangramExampleMap" \  
  --configuration "Style=VectorHereContrast"
```

### ℹ Note

La fatturazione è determinata dall'utilizzo da parte dell'utente. È possibile che vengano addebitati costi per l'utilizzo di altri servizi. AWS Per ulteriori informazioni, consulta i [prezzi di Amazon Location Service](#).

### Argomenti

- [Usare Tangram con Amazon Location Service](#)
- [Utilizzo di Tangram ES per Android con Amazon Location Service](#)
- [Utilizzo di Tangram ES per iOS con Amazon Location Service](#)

### Usare Tangram con Amazon Location Service

[Tangram](#) è un motore di mappatura flessibile, progettato per il rendering in tempo reale di mappe 2D e 3D da riquadri vettoriali. Può essere utilizzato con gli stili progettati da Mapzen e i riquadri HERE forniti dall'API Maps di Amazon Location Service. Questa guida descrive come integrare Tangram con Amazon Location all'interno di un'JavaScript applicazione HTML/ di base, sebbene le stesse librerie e tecniche si applichino anche quando si utilizzano framework come React e Angular.

Tangram è basato su [Leaflet](#), una libreria open source per mappe interattive ottimizzate per dispositivi mobili. JavaScript Ciò significa che molti plugin e controlli compatibili con Leaflet funzionano anche con Tangram.

Gli stili Tangram creati per funzionare con lo [schema Tilezen](#) sono ampiamente compatibili con Amazon Location quando si utilizzano mappe di HERE. Ciò include:

- [Bubble Wrap](#): uno stile di orientamento completo con icone utili per i punti di interesse
- [Cinnabar](#): un look classico e ideale per le applicazioni di mappatura generali
- [Refill](#): uno stile di mappa minimalista progettato per sovrapposizioni di visualizzazione dei dati, ispirato allo stile fondamentale Toner di Stamen Design
- [Tron](#) — Un'esplorazione delle trasformazioni di scala nel linguaggio visivo di TRON
- [Walkabout](#): uno stile incentrato sulle attività all'aperto, perfetto per le escursioni o per uscire

[Questa guida descrive come integrare Tangram con Amazon Location all'interno di un'JavaScript applicazione HTML/HTML di base utilizzando lo stile Tangram chiamato Bubble Wrap.](#) Questo esempio è disponibile come parte dell'archivio di esempi di Amazon Location Service su [GitHub](#).

Sebbene altri stili Tangram siano meglio accompagnati da riquadri raster, che codificano le informazioni sul terreno, questa funzionalità non è ancora supportata da Amazon Location.

#### Important

Gli stili Tangram illustrati nel seguente tutorial sono compatibili solo con le risorse della mappa di Amazon Location configurate con lo `VectorHereContrast` stile.

## Creazione dell'applicazione: Scaffolding

L'applicazione è una pagina HTML con JavaScript cui creare la mappa sulla tua applicazione web. Crea una pagina HTML (`index.html`) e crea il contenitore della mappa:

- Inserisci un `div` elemento con una `id` mappa per applicare le dimensioni della mappa alla visualizzazione della mappa.
- Le dimensioni vengono ereditate dalla finestra.

```
<html>
```

```
<head>
  <style>
    body {
      margin: 0;
    }

    #map {
      height: 100vh; /* 100% of viewport height */
    }
  </style>
</head>
<body>
  <!-- map container -->
  <div id="map" />
</body>
</html>
```

## Creazione dell'applicazione: aggiunta di dipendenze

Aggiungi le seguenti dipendenze:

- Leaflet e il relativo CSS associato.
- Tangram.
- SDK AWS per JavaScript.

```
<!-- CSS dependencies -->
<link
  rel="stylesheet"
  href="https://unpkg.com/leaflet@1.7.1/dist/leaflet.css"
  integrity="sha512-xodZBNTC5n17Xt2atTPuE1HxjVMSvLVW9ocqUKLsCC5CXdbqCmblAsh0MAS6/keqq/
sMZM19scR4PsZChSR7A=="
  crossorigin=""
/>
<!-- JavaScript dependencies -->
<script src="https://unpkg.com/leaflet@1.7.1/dist/leaflet.js"></script>
<script src="https://unpkg.com/tangram"></script>
<script src="https://sdk.amazonaws.com/js/aws-sdk-2.784.0.min.js"></script>
<script>
  // application-specific code
</script>
```

Questo crea una pagina vuota con i prerequisiti necessari. Il passaggio successivo ti guida nella scrittura del JavaScript codice per la tua applicazione.

## Creazione dell'applicazione: configurazione

Per configurare l'applicazione con le tue risorse e credenziali:

1. Inserisci i nomi e gli identificatori delle tue risorse.

```
// Cognito Identity Pool ID
const identityPoolId = "us-east-1:54f2ba88-9390-498d-aaa5-0d97fb7ca3bd";
// Amazon Location Service map name; must be HERE-backed
const mapName = "TangramExampleMap";
```

2. Crea un'istanza di un provider di credenziali utilizzando il pool di identità non autenticato che hai creato in [Uso delle mappe - Passaggio 2](#), Configurazione dell'autenticazione. Poiché utilizza credenziali al di fuori del normale flusso di lavoro dell'SDK AWS, le sessioni scadono dopo un'ora.

```
// extract the region from the Identity Pool ID; this will be used for both Amazon
  Cognito and Amazon Location
AWS.config.region = identityPoolId.split(":", 1)[0];

// instantiate a Cognito-backed credential provider
const credentials = new AWS.CognitoIdentityCredentials({
  IdentityPoolId: identityPoolId,
});
```

3. Sebbene Tangram consenta di sovrascrivere gli URL utilizzati per recuperare i riquadri, non include la possibilità di intercettare le richieste in modo che possano essere firmate.

Per ovviare a questo problema, esegui l'override in modo da `sources.mapzen.url` puntare ad Amazon Location utilizzando un nome host sintetico `amazon.location`, che verrà gestito da un addetto all'[assistenza](#). Di seguito è riportato un esempio di configurazione della scena con [Bubble Wrap](#):

```
const scene = {
  import: [
    // Bubble Wrap style
    "https://www.nextzen.org/carto/bubble-wrap-style/10/bubble-wrap-style.zip",
    "https://www.nextzen.org/carto/bubble-wrap-style/10/themes/label-7.zip",
```

```

    "https://www.nextzen.org/carto/bubble-wrap-style/10/themes/bubble-wrap-road-
shields-usa.zip",
    "https://www.nextzen.org/carto/bubble-wrap-style/10/themes/bubble-wrap-road-
shields-international.zip",
  ],
  // override values beneath the `sources` key in the style above
  sources: {
    mapzen: {
      // point at Amazon Location using a synthetic URL, which will be handled by
the service
      // worker
      url: `https://amazon.location/${mapName}/{z}/{x}/{y}`,
    },
    // effectively disable raster tiles containing encoded normals
    normals: {
      max_zoom: 0,
    },
    "normals-elevation": {
      max_zoom: 0,
    },
  },
},
};

```

## Creazione dell'applicazione: richiedi la trasformazione

Per registrare e inizializzare l'addetto all'assistenza, crea una `registerServiceWorker` funzione da chiamare prima dell'inizializzazione della mappa. Questo registra il JavaScript codice fornito in un file separato chiamato `service sw.js worker control. index.html`

Le credenziali vengono caricate da Amazon Cognito e passate al service worker insieme alla Regione per fornire informazioni per firmare le richieste di tile [con Signature](#) Version 4.

```

/**
 * Register a service worker that will rewrite and sign requests using Signature
Version 4.
 */
async function registerServiceWorker() {
  if ("serviceWorker" in navigator) {
    try {
      const reg = await navigator.serviceWorker.register("./sw.js");

      // refresh credentials from Amazon Cognito

```

```
await credentials.refreshPromise();

await reg.active.ready;

if (navigator.serviceWorker.controller == null) {
  // trigger a navigate event to active the controller for this page
  window.location.reload();
}

// pass credentials to the service worker
reg.active.postMessage({
  credentials: {
    accessKeyId: credentials.accessKeyId,
    secretAccessKey: credentials.secretAccessKey,
    sessionToken: credentials.sessionToken,
  },
  region: AWS.config.region,
});
} catch (error) {
  console.error("Service worker registration failed:", error);
}
} else {
  console.warn("Service worker support is required for this example");
}
}
```

L'implementazione di Service Worker in `sw.js` ascolta message gli eventi per rilevare le modifiche alle credenziali e alla configurazione della regione. Funziona anche come server proxy ascoltando gli fetch eventi. fetch gli eventi che hanno come target il nome host `amazon.location` sintetico verranno riscritti per indirizzare l'API Amazon Location appropriata e firmati utilizzando Amplify Core. Signer

```
// sw.js
self.importScripts(
  "https://unpkg.com/@aws-amplify/core@3.7.0/dist/aws-amplify-core.min.js"
);

const { Signer } = aws_amplify_core;

let credentials;
let region;

self.addEventListener("install", (event) => {
```

```
// install immediately
event.waitUntil(self.skipWaiting());
});

self.addEventListener("activate", (event) => {
  // control clients ASAP
  event.waitUntil(self.clients.claim());
});

self.addEventListener("message", (event) => {
  const {
    data: { credentials: newCredentials, region: newRegion },
  } = event;

  if (newCredentials !== null) {
    credentials = newCredentials;
  }

  if (newRegion !== null) {
    region = newRegion;
  }
});

async function signedFetch(request) {
  const url = new URL(request.url);
  const path = url.pathname.slice(1).split("/");

  // update URL to point to Amazon Location
  url.pathname = `/maps/v0/maps/${path[0]}/tiles/${path.slice(1).join("/")}`;
  url.host = `maps.geo.${region}.amazonaws.com`;
  // strip params (Tangram generates an empty api_key param)
  url.search = "";

  const signed = Signer.signUrl(url.toString(), {
    access_key: credentials.accessKeyId,
    secret_key: credentials.secretAccessKey,
    session_token: credentials.sessionToken,
  });

  return fetch(signed);
}

self.addEventListener("fetch", (event) => {
  const { request } = event;
```



```
// match the synthetic hostname we're telling Tangram to use
if (request.url.includes("amazon.location")) {
  return event.respondWith(signedFetch(request));
}

// fetch normally
return event.respondWith(fetch(request));
});
```

Per rinnovare automaticamente le credenziali e inviarle all'addetto all'assistenza prima della scadenza, utilizza la seguente funzione all'interno di: `index.html`

```
async function refreshCredentials() {
  await credentials.refreshPromise();

  if ("serviceWorker" in navigator) {
    const controller = navigator.serviceWorker.controller;

    controller.postMessage({
      credentials: {
        accessKeyId: credentials.accessKeyId,
        secretAccessKey: credentials.secretAccessKey,
        sessionToken: credentials.sessionToken,
      },
    });
  } else {
    console.warn("Service worker support is required for this example.");
  }

  // schedule the next credential refresh when they're about to expire
  setTimeout(refreshCredentials, credentials.expireTime - new Date());
}
```

### Creazione dell'applicazione: inizializzazione della mappa

Affinché la mappa venga visualizzata dopo il caricamento della pagina, è necessario inizializzare la mappa. È possibile modificare la posizione iniziale della mappa, aggiungere controlli aggiuntivi e sovrapporre i dati.

### Note

È necessario fornire un marchio denominativo o un'attribuzione di testo per ogni fornitore di dati che si utilizza, nell'applicazione o nella documentazione. Le stringhe di attribuzione sono incluse nella risposta del descrittore di stile sotto i `tastisources.esri.attribution`, `sources.here.attribution` e `source.grabmaptiles.attribution`. Poiché Tangram non richiede queste risorse ed è compatibile solo con le mappe di HERE, usa «© 2020 HERE». Quando utilizzi le risorse di Amazon Location con [fornitori di dati](#), assicurati di leggere i [termini e le condizioni del servizio](#).

```
/**
 * Initialize a map.
 */
async function initializeMap() {
  // register the service worker to handle requests to https://amazon.location
  await registerServiceWorker();

  // Initialize the map
  const map = L.map("map").setView([49.2819, -123.1187], 10);
  Tangram.leafletLayer({
    scene,
  }).addTo(map);
  map.attributionControl.setPrefix("");
  map.attributionControl.addAttribution("© 2020 HERE");
}

initializeMap();
```

## Esecuzione dell'applicazione

Per eseguire questo esempio, puoi:

- Usa un host che supporti HTTPS,
- Utilizza un server Web locale per rispettare le restrizioni di sicurezza degli addetti all'assistenza.

Per utilizzare un server web locale, puoi usare `npx`, perché è installato come parte di Node.js. È possibile utilizzare `npx serve` dall'interno della stessa directory di `index.html` e `esw.js`. Questo serve l'applicazione su [localhost:5000](http://localhost:5000).

## Il seguente è il file: index.html

```
<!-- index.html -->
<html>
  <head>
    <link
      rel="stylesheet"
      href="https://unpkg.com/leaflet@1.7.1/dist/leaflet.css"
      integrity="sha512-xodZBNTC5n17Xt2atTPuE1HxjVMSvLVW9ocqUKLsCC5CXdbqCmblAsh0MAS6/
keqq/sMzMZ19scR4PsZChSR7A=="
      crossorigin=""
    />
    <style>
      body {
        margin: 0;

        #map {
          height: 100vh;
        }
      </style>
    </head>

    <body>
      <div id="map" />
      <script src="https://unpkg.com/leaflet@1.7.1/dist/leaflet.js"></script>
      <script src="https://unpkg.com/tangram"></script>
      <script src="https://sdk.amazonaws.com/js/aws-sdk-2.784.0.min.js"></script>
      <script>
        // configuration
        // Cognito Identity Pool ID
        const identityPoolId = "<Identity Pool ID>";
        // Amazon Location Service Map name; must be HERE-backed
        const mapName = "<Map name>";

        AWS.config.region = identityPoolId.split(":")[0];

        // instantiate a credential provider
        credentials = new AWS.CognitoIdentityCredentials({
          IdentityPoolId: identityPoolId,
        });

        const scene = {
          import: [
```

```
// Bubble Wrap style
"https://www.nextzen.org/carto/bubble-wrap-style/10/bubble-wrap-style.zip",
"https://www.nextzen.org/carto/bubble-wrap-style/10/themes/label-7.zip",
"https://www.nextzen.org/carto/bubble-wrap-style/10/themes/bubble-wrap-road-
shields-usa.zip",
"https://www.nextzen.org/carto/bubble-wrap-style/10/themes/bubble-wrap-road-
shields-international.zip",
],
// override values beneath the `sources` key in the style above
sources: {
  mapzen: {
    // point at Amazon Location using a synthetic URL, which will be handled by
the service
    // worker
    url: `https://amazon.location/${mapName}/{z}/{x}/{y}`,
  },
  // effectively disable raster tiles containing encoded normals
  normals: {
    max_zoom: 0,
  },
  "normals-elevation": {
    max_zoom: 0,
  },
},
};

/**
 * Register a service worker that will rewrite and sign requests using Signature
Version 4.
 */
async function registerServiceWorker() {
  if ("serviceWorker" in navigator) {
    try {
      const reg = await navigator.serviceWorker.register("./sw.js");

      // refresh credentials from Amazon Cognito
      await credentials.refreshPromise();

      await reg.active.ready;

      if (navigator.serviceWorker.controller == null) {
        // trigger a navigate event to active the controller for this page
        window.location.reload();
      }
    }
  }
}
```

```
// pass credentials to the service worker
reg.active.postMessage({
  credentials: {
    accessKeyId: credentials.accessKeyId,
    secretAccessKey: credentials.secretAccessKey,
    sessionToken: credentials.sessionToken,
  },
  region: AWS.config.region,
});
} catch (error) {
  console.error("Service worker registration failed:", error);
}
} else {
  console.warn("Service Worker support is required for this example");
}
}

/**
 * Initialize a map.
 */
async function initializeMap() {
  // register the service worker to handle requests to https://amazon.location
  await registerServiceWorker();

  // Initialize the map
  const map = L.map("map").setView([49.2819, -123.1187], 10);
  Tangram.leafletLayer({
    scene,
  }).addTo(map);
  map.attributionControl.setPrefix("");
  map.attributionControl.addAttribution("© 2020 HERE");
}

initializeMap();
</script>
</body>
</html>
```

Il seguente è il `sw.js` file:

```
// sw.js
self.importScripts(
```

```
"https://unpkg.com/@aws-amplify/core@3.7.0/dist/aws-amplify-core.min.js"
);

const { Signer } = aws_amplify_core;

let credentials;
let region;

self.addEventListener("install", (event) => {
  // install immediately
  event.waitUntil(self.skipWaiting());
});

self.addEventListener("activate", (event) => {
  // control clients ASAP
  event.waitUntil(self.clients.claim());
});

self.addEventListener("message", (event) => {
  const {
    data: { credentials: newCredentials, region: newRegion },
  } = event;

  if (newCredentials !== null) {
    credentials = newCredentials;
  }

  if (newRegion !== null) {
    region = newRegion;
  }
});

async function signedFetch(request) {
  const url = new URL(request.url);
  const path = url.pathname.slice(1).split("/");

  // update URL to point to Amazon Location
  url.pathname = `/maps/v0/maps/${path[0]}/tiles/${path.slice(1).join("/")}`;
  url.host = `maps.geo.${region}.amazonaws.com`;
  // strip params (Tangram generates an empty api_key param)
  url.search = "";

  const signed = Signer.signUrl(url.toString(), {
    access_key: credentials.accessKeyId,
```

```
    secret_key: credentials.secretAccessKey,
    session_token: credentials.sessionToken,
  });

  return fetch(signed);
}

self.addEventListener("fetch", (event) => {
  const { request } = event;

  // match the synthetic hostname we're telling Tangram to use
  if (request.url.includes("amazon.location")) {
    return event.respondWith(signedFetch(request));
  }

  // fetch normally
  return event.respondWith(fetch(request));
});
```

Questo esempio è disponibile come parte dell'archivio di esempi di Amazon Location Service su [GitHub](#).

### Utilizzo di Tangram ES per Android con Amazon Location Service

[Tangram ES](#) è una libreria C++ per il rendering di mappe 2D e 3D da dati vettoriali utilizzando OpenGL ES. [È la controparte nativa di Tangram.](#)

Gli stili Tangram creati per funzionare con lo [schema Tilezen](#) sono ampiamente compatibili con Amazon Location quando si utilizzano mappe di HERE. Ciò include:

- [Bubble Wrap](#): uno stile di orientamento completo con icone utili per i punti di interesse.
- [Cinnabar](#): un look classico e ideale per le applicazioni di mappatura generali.
- [Refill](#): uno stile di mappa minimalista progettato per sovrapposizioni di visualizzazione dei dati, ispirato allo stile fondamentale Toner di Stamen Design.
- [Tron](#) — Un'esplorazione delle trasformazioni di scala nel linguaggio visivo di TRON.
- [Walkabout](#): uno stile incentrato sull'outdoor, perfetto per le escursioni o per uscire.

Questa guida descrive come integrare Tangram ES per Android con Amazon Location utilizzando lo stile Tangram chiamato Cinnabar. Questo esempio è disponibile come parte dell'archivio di esempi di Amazon Location Service su [GitHub](#).

Sebbene altri stili Tangram siano meglio accompagnati da riquadri raster, che codificano le informazioni sul terreno, questa funzionalità non è ancora supportata da Amazon Location.

### Important

Gli stili Tangram illustrati nel seguente tutorial sono compatibili solo con le risorse della mappa di Amazon Location configurate con lo `VectorHereContrast` stile.

## Creazione dell'applicazione: inizializzazione

Per inizializzare l'applicazione:

1. Crea un nuovo progetto Android Studio dal modello Empty Activity.
2. Assicurati che Kotlin sia selezionato per la lingua del progetto.
3. Seleziona un SDK minimo di API 16: Android 4.1 (Jelly Bean) o versione successiva.
4. Apri la struttura del progetto per selezionare File, Struttura del progetto... e scegli la sezione Dipendenze.
5. Una <All Modules>volta selezionata, scegliete il pulsante + per aggiungere una nuova dipendenza dalla libreria.
6. Aggiungi AWS Android SDK versione 2.19.1 o successiva. Ad esempio: `com.amazonaws:aws-android-sdk-core:2.19.1`
7. Aggiungi Tangram versione 0.13.0 o successiva. Ad esempio: `com.mapzen.tangram:tangram:0.13.0`.

### Note

Cercando Tangram: `com.mapzen.tangram:tangram:0.13.0` genererà un messaggio che indica che «non è stato trovato», ma scegliendo OK ne consentirà l'aggiunta.

## Creazione dell'applicazione: configurazione

Per configurare l'applicazione con le tue risorse e la tua AWS regione:

1. Creare il `app/src/main/res/values/configuration.xml`.



- Inserisci i nomi e gli identificatori delle tue risorse e anche la AWS regione in cui sono state create:

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
  <string name="identityPoolId">us-east-1:54f2ba88-9390-498d-aaa5-0d97fb7ca3bd</string>
  <string name="mapName">TangramExampleMap</string>
  <string name="awsRegion">us-east-1</string>
  <string name="sceneUrl">https://www.nextzen.org/carto/cinnabar-style/9/cinnabar-style.zip</string>
  <string name="attribution">© 2020 HERE</string>
</resources>
```

Creazione dell'applicazione: Layout delle attività

Modifica `app/src/main/res/layout/activity_main.xml`:

- Aggiungi un `MapView`, che rende la mappa. Questo imposterà anche il punto centrale iniziale della mappa.
- Aggiungi un `TextView`, che mostra l'attribuzione.

Questo imposterà anche il punto centrale iniziale della mappa.

### Note

È necessario fornire un marchio denominativo o un'attribuzione di testo per ogni fornitore di dati che si utilizza, nell'applicazione o nella documentazione. Le stringhe di attribuzione sono incluse nella risposta del descrittore di stile sotto i `tastisources.esri.attribution`, `sources.here.attribution` e `source.grabmaptiles.attribution`. Poiché Tangram non richiede queste risorse ed è compatibile solo con le mappe di HERE, usa «© 2020 HERE». Quando utilizzi le risorse di Amazon Location con [fornitori di dati](#), assicurati di leggere i [termini e le condizioni del servizio](#).

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
  xmlns:android="http://schemas.android.com/apk/res/android"
```

```
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
tools:context=".MainActivity">

<com.mapzen.tangram.MapView
    android:id="@+id/map"
    android:layout_height="match_parent"
    android:layout_width="match_parent" />

<TextView
    android:id="@+id/attributionView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:background="#80808080"
    android:padding="5sp"
    android:textColor="@android:color/black"
    android:textSize="10sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    tools:ignore="SmallSp" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

Creazione dell'applicazione: richiedi la trasformazione

Crea una classe denominata `SigV4Interceptor` per intercettare AWS le richieste e firmarle utilizzando [Signature Version 4](#). Questo verrà registrato con il client HTTP utilizzato per recuperare le risorse della mappa quando viene creata l'attività principale.

```
package aws.location.demo.okhttp

import com.amazonaws.DefaultRequest
import com.amazonaws.auth.AWS4Signer
import com.amazonaws.auth.AWSCredentialsProvider
import com.amazonaws.http.HttpMethodName
import com.amazonaws.util.IOUtils
import okhttp3.HttpUrl
import okhttp3.Interceptor
import okhttp3.Request
import okhttp3.Response
import okio.Buffer
import java.io.ByteArrayInputStream
```

```
import java.net.URI

class SigV4Interceptor(
    private val credentialsProvider: AWSCredentialsProvider,
    private val serviceName: String
) : Interceptor {
    override fun intercept(chain: Interceptor.Chain): Response {
        val originalRequest = chain.request()

        if (originalRequest.url().host().contains("amazonaws.com")) {
            val signer = if (originalRequest.url().encodedPath().contains("@")) {
                // the presence of "@" indicates that it doesn't need to be double URL-
                encoded
                AWS4Signer(false)
            } else {
                AWS4Signer()
            }

            val awsRequest = toAWSRequest(originalRequest, serviceName)
            signer.setServiceName(serviceName)
            signer.sign(awsRequest, credentialsProvider.credentials)

            return chain.proceed(toSignedOkHttpRequest(awsRequest, originalRequest))
        }

        return chain.proceed(originalRequest)
    }

    companion object {
        fun toAWSRequest(request: Request, serviceName: String): DefaultRequest<Any> {
            // clone the request (AWS-style) so that it can be populated with
            credentials
            val dr = DefaultRequest<Any>(serviceName)

            // copy request info
            dr.httpMethod = HttpMethodName.valueOf(request.method())
            with(request.url()) {
                dr.resourcePath = uri().path
                dr.endpoint = URI.create("${scheme()}://${host()}")

                // copy parameters
                for (p in queryParameterNames()) {
                    if (p != "") {
                        dr.addParameter(p, queryParameter(p))
                    }
                }
            }
        }
    }
}
```

```
        }
    }
}

// copy headers
for (h in request.headers().names()) {
    dr.addHeader(h, request.header(h))
}

// copy the request body
val bodyBytes = request.body()?.let { body ->
    val buffer = Buffer()
    body.writeTo(buffer)
    IOUtils.toByteArray(buffer.inputStream())
}

dr.content = ByteArrayInputStream(bodyBytes ?: ByteArray(0))

return dr
}

fun toSignedOkHttpRequest(
    awsRequest: DefaultRequest<Any>,
    originalRequest: Request
): Request {
    // copy signed request back into an OkHttp Request
    val builder = Request.Builder()

    // copy headers from the signed request
    for ((k, v) in awsRequest.headers) {
        builder.addHeader(k, v)
    }

    // start building an HttpUrl
    val urlBuilder = HttpUrl.Builder()
        .host(awsRequest.endpoint.host)
        .scheme(awsRequest.endpoint.scheme)
        .encodedPath(awsRequest.resourcePath)

    // copy parameters from the signed request
    for ((k, v) in awsRequest.parameters) {
        urlBuilder.addQueryParameter(k, v)
    }
}
```

```
        return builder.url(urlBuilder.build())
            .method(originalRequest.method(), originalRequest.body())
            .build()
    }
}
```

## Creazione dell'applicazione: attività principale

L'attività principale è responsabile dell'inizializzazione delle visualizzazioni che verranno mostrate agli utenti. Ciò comporta:

- Creazione di un'istanza di Amazon Cognito. `CredentialsProvider`
- Registrazione dell'interceptor `Signature Version 4`.
- Configurazione della mappa puntandola su uno stile di mappa, sovrascrivendo gli URL dei riquadri e visualizzando l'attribuzione appropriata.

`MainActivity` è anche responsabile dell'inoltro degli eventi del ciclo di vita alla visualizzazione della mappa.

```
package aws.location.demo.tangram

import android.os.Bundle
import android.widget.TextView
import androidx.appcompat.app.AppCompatActivity
import aws.location.demo.okhttp.SigV4Interceptor
import com.amazonaws.auth.CognitoCachingCredentialsProvider
import com.amazonaws.regions.Regions
import com.mapzen.tangram.*
import com.mapzen.tangram.networking.DefaultHttpHandler
import com.mapzen.tangram.networking.HttpHandler

private const val SERVICE_NAME = "geo"

class MainActivity : AppCompatActivity(), MapView.MapReadyCallback {
    private var mapView: MapView? = null

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)

        setContentView(R.layout.activity_main)
    }
}
```

```
        mapView = findViewById(R.id.map)

        mapView?.getMapAsync(this, getHttpHandler())
        findViewById<TextView>(R.id.attributionView).text =
getString(R.string.attribution)
    }

    override fun onMapReady(mapController: MapController?) {
        val sceneUpdates = arrayListOf(
            SceneUpdate(
                "sources.mapzen.url",
                "https://maps.geo.${getString(R.string.awsRegion)}.amazonaws.com/maps/
v0/maps/${
                    getString(
                        R.string.mapName
                    )
                }/tiles/{z}/{x}/{y}"
            )
        )

        mapController?.let { map ->
            map.updateCameraPosition(
                CameraUpdateFactory.newLngLatZoom(
                    LngLat(-123.1187, 49.2819),
                    12F
                )
            )
            map.loadSceneFileAsync(
                getString(R.string.sceneUrl),
                sceneUpdates
            )
        }
    }

    private fun getHttpHandler(): HttpHandler {
        val builder = DefaultHttpHandler.getClientBuilder()

        val credentialsProvider = CognitoCachingCredentialsProvider(
            applicationContext,
            getString(R.string.identityPoolId),
            Regions.US_EAST_1
        )
    }
}
```

```
        return DefaultHttpHandler(
            builder.addInterceptor(
                SigV4Interceptor(
                    credentialsProvider,
                    SERVICE_NAME
                )
            )
        )
    }

    override fun onResume() {
        super.onResume()
        mapView?.onResume()
    }

    override fun onPause() {
        super.onPause()
        mapView?.onPause()
    }

    override fun onLowMemory() {
        super.onLowMemory()
        mapView?.onLowMemory()
    }

    override fun onDestroy() {
        super.onDestroy()
        mapView?.onDestroy()
    }
}
```

L'esecuzione di questa applicazione consente di visualizzare una mappa a schermo intero nello stile desiderato. Questo esempio è disponibile come parte dell'archivio di esempi di Amazon Location Service su [GitHub](#).

### Utilizzo di Tangram ES per iOS con Amazon Location Service


[Tangram ES](#) è una libreria C++ per il rendering di mappe 2D e 3D da dati vettoriali utilizzando OpenGL ES. [È la controparte nativa di Tangram.](#)

Gli stili Tangram creati per funzionare con lo [schema Tilezen](#) sono ampiamente compatibili con Amazon Location quando si utilizzano mappe di HERE. Ciò include:

- [Bubble Wrap](#): uno stile di orientamento completo con icone utili per i punti di interesse
- [Cinnabar](#): un look classico e ideale per le applicazioni di mappatura generali
- [Refill](#): uno stile di mappa minimalista progettato per sovrapposizioni di visualizzazione dei dati, ispirato allo stile fondamentale Toner di Stamen Design
- [Tron](#) — Un'esplorazione delle trasformazioni di scala nel linguaggio visivo di TRON
- [Walkabout](#): uno stile incentrato sulle attività all'aperto, perfetto per le escursioni o per uscire

Questa guida descrive come integrare Tangram ES per iOS con Amazon Location utilizzando lo stile Tangram chiamato Cinnabar. Questo esempio è disponibile come parte dell'archivio di esempi di Amazon Location Service su [GitHub](#).

Sebbene altri stili Tangram siano meglio accompagnati da riquadri raster, che codificano le informazioni sul terreno, questa funzionalità non è ancora supportata da Amazon Location.

 Important

Gli stili Tangram illustrati nel seguente tutorial sono compatibili solo con le risorse della mappa di Amazon Location configurate con lo `VectorHereContrast` stile.

## Creazione dell'applicazione: inizializzazione

Per inizializzare l'applicazione:

1. Crea un nuovo progetto Xcode dal modello dell'app.
2. Seleziona SwiftUI per la sua interfaccia.
3. Seleziona l'applicazione SwiftUI per il suo ciclo di vita.
4. Seleziona Swift per la sua lingua.

## Creazione dell'applicazione: aggiungi dipendenze

Per aggiungere dipendenze, puoi utilizzare un gestore delle dipendenze, ad esempio: [CocoaPods](#)

1. Nel tuo terminale, installa: CocoaPods

```
sudo gem install cocoapods
```



2. Vai alla directory del progetto dell'applicazione e inicializza il Podfile con il gestore di CocoaPods pacchetti:

```
pod init
```

3. Apri il Podfile per aggiungere AWSCore e Tangram-es come dipendenze:

```
platform :ios, '12.0'

target 'Amazon Location Service Demo' do
  use_frameworks!

  pod 'AWSCore'
  pod 'Tangram-es'
end
```

4. Scarica e installa le dipendenze:

```
pod install --repo-update
```

5. Apri lo spazio di lavoro Xcode che ha creato: CocoaPods

```
xed .
```

## Creazione dell'applicazione: configurazione

Aggiungi le seguenti chiavi e valori a Info.plist per configurare l'applicazione e disabilitare la telemetria:

Chiave	Valore
AWSRegion	us-east-1
IdentityPoolId	us-east- 1:54 f2ba88-9390-498d-aaa5-0d97f b7ca3bd
MapName	ExampleMap
URL della scena	<a href="https://www.nextzen.org/carto/cinnabar-style/9/cinnabar-style.zip">https://www.nextzen.org/carto/cinnabar-style/9/cinnabar-style.zip</a>

## Creazione dell'applicazione: ContentView layout

Per renderizzare la mappa, modifica `ContentView.swift`:

- Aggiungi un file `MapView` che renderizza la mappa.
- Aggiungi un `TextField` che mostra l'attribuzione.

Questo imposta anche il punto centrale iniziale della mappa.

### Note

È necessario fornire un marchio denominativo o un'attribuzione di testo per ogni fornitore di dati che si utilizza, nell'applicazione o nella documentazione. Le stringhe di attribuzione sono incluse nella risposta del descrittore di stile sotto i tasti `sources.esri.attribution`, `sources.here.attribution` e `source.grabmaptiles.attribution`. Quando utilizzi le risorse di Amazon Location con [fornitori di dati](#), assicurati di leggere i [termini e le condizioni del servizio](#).

```
import SwiftUI
import TangramMap

struct ContentView: View {
    var body: some View {
        MapView()
            .cameraPosition(TGCameraPosition(
                center: CLLocationCoordinate2DMake(49.2819, -123.1187),
                zoom: 10,
                bearing: 0,
                pitch: 0))
            .edgesIgnoringSafeArea(.all)
            .overlay(
                Text("© 2020 HERE")
                    .disabled(true)
                    .font(.system(size: 12, weight: .light, design: .default))
                    .foregroundColor(.black)
                    .background(Color.init(Color.RGBColorSpace.sRGB, white: 0.5,
opacity: 0.5))
                    .cornerRadius(1),
                alignment: .bottomTrailing)
```

```

    }
}

struct ContentView_Previews: PreviewProvider {
    static var previews: some View {
        ContentView()
    }
}

```

Creazione dell'applicazione: richiedi la trasformazione

Crea un nuovo file Swift denominato `AWSSignatureV4URLHandler.swift` contenente la seguente definizione di classe per intercettare AWS le richieste e firmarle utilizzando [Signature](#) Version 4. Questo verrà registrato come gestore di URL all'interno di `Tangram.MapView`

```

import AWSCore
import TangramMap

class AWSSignatureV4URLHandler: TGDefaultURLHandler {
    private let region: AWSRegionType
    private let identityPoolId: String
    private let credentialsProvider: AWSCredentialsProvider

    init(region: AWSRegionType, identityPoolId: String) {
        self.region = region
        self.identityPoolId = identityPoolId
        self.credentialsProvider = AWSCognitoCredentialsProvider(regionType: region,
identityPoolId: identityPoolId)
        super.init()
    }

    override func downloadRequestAsync(_ url: URL, completionHandler: @escaping
TGDownloadCompletionHandler) -> UInt {
        if url.host?.contains("amazonaws.com") != true {
            // not an AWS URL
            return super.downloadRequestAsync(url, completionHandler:
completionHandler)
        }

        // URL-encode spaces, etc.
        let keyPath = String(url.path.dropFirst())
        guard let keyPathSafe =
keyPath.addingPercentEncoding(withAllowedCharacters: .urlPathAllowed) else {

```

```

        print("Invalid characters in path '\(keyPath)'; unsafe to sign")
        return super.downloadRequestAsync(url, completionHandler:
completionHandler)
    }

    // sign the URL
    let endpoint = AWSEndpoint(region: region, serviceName: "geo", url: url)
    let requestHeaders: [String: String] = ["host": endpoint!.hostName]
    let task = AWSSignatureV4Signer
        .generateQueryStringForSignatureV4(
            withCredentialProvider: credentialsProvider,
            httpMethod: .GET,
            expireDuration: 60,
            endpoint: endpoint!,
            keyPath: keyPathSafe,
            requestHeaders: requestHeaders,
            requestParameters: .none,
            signBody: true)
    task.waitUntilFinished()

    if let error = task.error as NSError? {
        print("Error occurred: \(error)")
    }

    if let result = task.result {
        // have Tangram fetch the signed URL
        return super.downloadRequestAsync(result as URL, completionHandler:
completionHandler)
    }

    // fall back to an unsigned URL
    return super.downloadRequestAsync(url, completionHandler: completionHandler)
}
}

```

## Creazione dell'applicazione: visualizzazione della mappa

La visualizzazione della mappa è responsabile dell'inizializzazione `AWSSignatureV4Delegate` e della configurazione di un'istanza di quella sottostante `MGLMapView`, che recupera le risorse e rende la mappa. Gestisce inoltre la propagazione delle stringhe di attribuzione dalla fonte del descrittore di stile al `ContentView`

Crea un nuovo file Swift denominato `MapView.swift` contenente la seguente definizione: `struct`

```
import AWSCore
import TangramMap
import SwiftUI

struct MapView: UIViewRepresentable {
    private let mapView: TGMapView

    init() {
        let regionName = Bundle.main.object(forKey: "AWSRegion") as!
String
        let identityPoolId = Bundle.main.object(forKey: "IdentityPoolId")
as! String
        let mapName = Bundle.main.object(forKey: "MapName") as! String
        let sceneURL = URL(string: Bundle.main.object(forKey: "SceneURL")
as! String)!

        let region = (regionName as NSString).aws_regionTypeValue()

        // rewrite tile URLs to point at AWS resources
        let sceneUpdates = [
            TGSceneUpdate(path: "sources.mapzen.url",
                value: "https://maps.geo.\(regionName).amazonaws.com/maps/v0/
maps/\(mapName)/tiles/{z}/{x}/{y}")]

        // instantiate a TGURLHandler that will sign AWS requests
        let urlHandler = AWSSignatureV4URLHandler(region: region, identityPoolId:
identityPoolId)

        // instantiate the map view and attach the URL handler
        mapView = TGMapView(frame: .zero, urlHandler: urlHandler)

        // load the map style and apply scene updates (properties modified at runtime)
        mapView.loadScene(from: sceneURL, with: sceneUpdates)
    }

    func cameraPosition(_ cameraPosition: TGCameraPosition) -> MapView {
        mapView.cameraPosition = cameraPosition

        return self
    }

    // MARK: - UIViewRepresentable protocol
```

```
func makeUIView(context: Context) -> TGMMapView {
    return mapView
}

func updateUIView(_ uiView: TGMMapView, context: Context) {
}
}
```

L'esecuzione di questa applicazione mostra una mappa a schermo intero nello stile che preferisci. Questo esempio è disponibile come parte dell'archivio di esempi di Amazon Location Service su [GitHub](#).

## Disegnare caratteristiche dei dati su una mappa

Dopo avere un'applicazione che esegue il rendering di una mappa, utilizzando Amplify o Tangram per renderizzare la mappa MapLibre, il passaggio successivo naturale è disegnare delle feature sulla parte superiore della mappa. Ad esempio, potresti voler visualizzare le posizioni dei clienti come indicatori sulla mappa.

In generale, puoi utilizzare le [funzioni di ricerca Luoghi](#) per trovare località dai tuoi dati, quindi utilizzare la funzionalità di Amplify MapLibre o Tangram per renderizzare le posizioni.

Per vedere esempi di rendering di diversi tipi di oggetti sulla mappa, consulta i seguenti esempi: MapLibre

- [Esempio: disegna dei marcatori](#)
- [Esempio: disegna punti raggruppati](#)
- [Esempio: disegnare un poligono](#)

Per altri esempi e tutorial, consulta [Esempi di codice e tutorial per lavorare con Amazon Location Service](#)

## Impostazione delle estensioni per una mappa utilizzando MapLibre

A volte non si desidera consentire agli utenti di eseguire la panoramica o lo zoom in tutto il mondo. Se utilizzi il controllo MapLibre della mappa, puoi limitare le estensioni o i limiti del controllo della mappa con l'opzione `maxBounds` e limitare lo zoom con `minZoom` e opzioni `maxZoom`.

Il seguente esempio di codice mostra come inizializzare il controllo della mappa per vincolare il panning a un limite specifico (in questo caso, le estensioni dell'origine dati Grab).

**Note**

Questi esempi sono inclusi e funzionano nel JavaScript contesto del tutorial. [Creazione di un'app Web](#)

```
// Set bounds to Grab data provider region
var bounds = [
    [90.0, -21.943045533438166], // Southwest coordinates
    [146.25, 31.952162238024968] // Northeast coordinates
];

var mlg1Map = new maplibregl.Map(
    {
        container: 'map',
        style: mapName,
        maxBounds: bounds // Sets bounds as max
        transformRequest,
    }
);
```

Allo stesso modo, è possibile impostare un livello di zoom minimo e massimo per la mappa. I valori per entrambi possono essere compresi tra 0 e 24, sebbene i valori predefiniti siano 0 per lo zoom minimo e 22 per il massimo (i fornitori di dati potrebbero non fornire dati a tutti i livelli di zoom). La maggior parte delle librerie di mappe lo gestisce automaticamente). L'esempio seguente inizializza le `maxZoom` opzioni `minZoom` and sul controllo MapLibre Map.

```
// Set the minimum and maximum zoom levels
var mlg1Map = new maplibregl.Map(
    {
        container: 'map',
        style: mapName,
        maxZoom: 12,
        minZoom: 5,
        transformRequest,
    }
);
```

**i** Tip

Il controllo MapLibre Map consente inoltre di impostare queste opzioni in fase di esecuzione, anziché durante l'inizializzazione, con `get . . .` le funzioni `and. set . . .`. Ad esempio, usa `getMaxBounds` e modifica `setMaxBounds` i limiti della mappa in fase di esecuzione.

## Gestione delle risorse cartografiche

Puoi gestire le risorse delle mappe utilizzando la console Amazon LocationAWS CLI, o le API di Amazon Location.

### Elenca le risorse della mappa

Puoi visualizzare un elenco delle tue risorse cartografiche utilizzando la console Amazon LocationAWS CLI, o le API Amazon Location.

#### Console

Per visualizzare un elenco di risorse cartografiche esistenti utilizzando la console Amazon Location

1. Apri la console Amazon Location all'[indirizzo https://console.aws.amazon.com/location/](https://console.aws.amazon.com/location/).
2. Scegli Mappe dal riquadro di navigazione a sinistra.
3. Visualizza un elenco delle risorse cartografiche in Le mie mappe.

#### API

Usa l'[ListMaps](#) operazione dalle API di Amazon Location Maps.

L'esempio seguente è una richiesta API per ottenere un elenco di risorse cartografiche nell'AWSaccount.

```
POST /maps/v0/list-maps
```

Di seguito è riportato un esempio di risposta per [ListMaps](#):

```
{
  "Entries": [
    {
```



```
    "CreateTime": 2020-10-30T01:38:36Z,  
    "DataSource": "Esri",  
    "Description": "string",  
    "MapName": "ExampleMap",  
    "UpdateTime": 2020-10-30T01:38:36Z  
  }  
],  
"NextToken": "1234-5678-9012"  
}
```

## CLI

Utilizza il comando [list-map](#).

L'esempio seguente è un AWS CLI modo per ottenere un elenco di risorse cartografiche nell'AWSaccount.

```
aws location list-maps
```

## Ottieni i dettagli delle risorse cartografiche

Puoi ottenere dettagli su qualsiasi risorsa cartografica nel tuo account AWS utilizzando la console Amazon LocationAWS CLI, o le API Amazon Location.

### Console

Per visualizzare i dettagli di una risorsa cartografica utilizzando la console Amazon Location

1. Apri la console Amazon Location all'[indirizzo https://console.aws.amazon.com/location/](https://console.aws.amazon.com/location/).
2. Scegli Mappe dal riquadro di navigazione a sinistra.
3. In Le mie mappe, seleziona il link del nome della risorsa della mappa di destinazione.

### API

Usa l'[DescribeMap](#)operazione dalle API di Amazon Location Maps.

L'esempio seguente è una richiesta API per cui ottenere i dettagli della risorsa cartografica.

*ExampleMap*

```
GET /maps/v0/maps/ExampleMap
```

Di seguito è riportato un esempio di risposta per [DescribeMap](#):

```
{
  "Configuration": {
    "Style": "VectorEsriNavigation"
  },
  "CreateTime": "2020-10-30T01:38:36Z",
  "DataSource": "Esri",
  "Description": "string",
  "MapArn": "arn:aws:geo:us-west-2:123456789012:maps/ExampleMap",
  "MapName": "ExampleMap",
  "Tags": {
    "Tag1" : "Value1"
  },
  "UpdateTime": "2020-10-30T01:40:36Z"
}
```

## CLI

Utilizza il comando [describe-map](#).

L'esempio seguente è un AWS CLI modo per ottenere i dettagli delle risorse della mappa per *ExampleMap*.

```
aws location describe-map \
  --map-name "ExampleMap"
```

## Eliminare una risorsa cartografica

Puoi eliminare una risorsa cartografica dal tuo AWS account utilizzando la console Amazon LocationAWS CLI, o le API Amazon Location.

### Warning

Questa operazione elimina la risorsa in modo permanente.

## Console

Per eliminare una risorsa cartografica esistente utilizzando la console Amazon Location

1. Apri la console Amazon Location all'[indirizzo https://console.aws.amazon.com/location/](https://console.aws.amazon.com/location/).
2. Scegli Mappe dal riquadro di navigazione a sinistra.
3. Nella sezione Le mie mappe, seleziona la mappa di destinazione dall'elenco.
4. Scegli Elimina mappa.

## API

Usa l'[DeleteMap](#) operazione dalle API di Amazon Location Maps.

L'esempio seguente è una richiesta API per eliminare la risorsa *ExampleMap* della mappa.

```
DELETE /maps/v0/maps/ExampleMap
```

Di seguito è riportato un esempio di risposta riuscita per [DeleteMap](#):

```
HTTP/1.1 200
```

## CLI

Utilizza il comando [delete-map](#).

L'esempio seguente è un AWS CLI comando per eliminare la risorsa della mappa *ExampleMap*.

```
aws location delete-map \  
  --map-name "ExampleMap"
```

## Ricerca di dati relativi a località e geolocalizzazione utilizzando Amazon Location

Amazon Location include la possibilità di cercare i dati di geolocalizzazione, o località, del provider scelto. Sono disponibili diversi tipi di ricerca.

- **Geocodifica:** la geocodifica è il processo di ricerca di indirizzi, aree geografiche, nomi commerciali o altri punti di interesse, in base all'immissione di testo. Restituisce i dettagli e la posizione (in latitudine e longitudine) dei risultati trovati.
- **Geocodifica inversa:** la geocodifica inversa consente di trovare luoghi vicini a una determinata posizione.

- **Completamento automatico:** il completamento automatico è il processo di creazione di suggerimenti automatici mentre l'utente digita una query. Ad esempio, se digitano **Par** un suggerimento potrebbe essere. `Paris, France`

Amazon Location ti consente di scegliere un fornitore di dati per le operazioni di ricerca dei luoghi creando e configurando una risorsa per l'indice dei luoghi.

Una volta creata la risorsa, puoi inviare richieste utilizzando l' AWS SDK per la tua lingua preferita, Amplify o gli endpoint dell'API REST. Puoi utilizzare i dati della risposta per contrassegnare le posizioni su una mappa, arricchire i dati di posizione e convertire le posizioni in testo leggibile dall'uomo.

#### Note

Per una panoramica dei concetti relativi ai luoghi di ricerca, consulta. [Ricerca di luoghi](#)

## Argomenti

- [Prerequisiti](#)
- [Geocodifica](#)
- [Geocodifica inversa](#)
- [Completamento automatico](#)
- [Utilizzo degli ID dei luoghi](#)
- [Posiziona le categorie e filtra i risultati](#)
- [Amazon Aurora PostgreSQLfunzioni definite dall'utente per Amazon Location Service](#)
- [Gestione delle risorse relative all'indice dei luoghi](#)

## Prerequisiti

Prima di iniziare la geocodifica, la geocodifica inversa o la ricerca di luoghi, segui i passaggi preliminari:

### Argomenti

- [Creazione di una risorsa per l'indice dei luoghi](#)
- [Autenticazione delle tue richieste](#)

## Creazione di una risorsa per l'indice dei luoghi

Inizia creando una risorsa dedicata all'indice dei luoghi nel tuo AWS account.

Quando crei una risorsa per l'indice dei luoghi, puoi scegliere tra i fornitori di dati disponibili per supportare le query per la geocodifica, la geocodifica inversa e le ricerche:

1. Esri: [per ulteriori informazioni sulla copertura di Esri nella regione di interesse, vedere la copertura di geocodifica Esri nella documentazione Esri.](#)
2. Tecnologie HERE — Per ulteriori informazioni sulla copertura di HERE nella tua regione di interesse, consulta la copertura di [geocodifica HERE nella documentazione HERE.](#)
3. Grab: Grab fornisce dati solo per il sud-est asiatico. Per ulteriori informazioni sulla copertura di Grab, consulta [Paesi/regioni e aree coperte](#) questa guida.

Puoi farlo utilizzando la console Amazon Location Service AWS CLI, o le API Amazon Location.

### Console

Per creare una risorsa relativa all'indice dei luoghi utilizzando la console Amazon Location Service

1. Apri la console Amazon Location Service all'[indirizzo https://console.aws.amazon.com/location/](https://console.aws.amazon.com/location/).
2. Nel riquadro di navigazione a sinistra, scegli Posizione indici.
3. Scegli Crea indice dei luoghi.
4. Compila le seguenti caselle:
  - Nome: inserisci un nome per la risorsa dell'indice dei luoghi. Ad esempio, *ExamplePlaceIndex*. Massimo 100 caratteri. Le voci valide includono caratteri alfanumerici, trattini, punti e caratteri di sottolineatura.
  - Descrizione: immettere una descrizione facoltativa.
5. In Fornitori di dati, scegli un [fornitore di dati](#) disponibile da utilizzare con la tua risorsa relativa all'indice dei luoghi.

#### Note

Se la tua applicazione esegue il tracciamento o l'instradamento di risorse utilizzate nella tua azienda, ad esempio veicoli per la consegna o dipendenti, non devi

utilizzare Esri come provider di geolocalizzazione. Per ulteriori dettagli, consulta la sezione 82 dei [Termini di servizio di AWS](#).

6. In Opzioni di archiviazione dei dati, specifica se intendi archiviare i risultati della ricerca dalla risorsa dell'indice dei luoghi.
7. (Facoltativo) In Tag, immettere Key (Chiave) e Value (Valore) di un tag. Questo aggiunge un tag alla nuova risorsa dell'indice dei luoghi. Per ulteriori informazioni, consulta [Tagging delle risorse](#).
8. Scegli Crea indice dei luoghi.

## API

Per creare una risorsa relativa all'indice dei luoghi utilizzando le API di Amazon Location

Usa l'[CreatePlaceIndex](#) operazione dalle API di Amazon Location Places.

L'esempio seguente è una richiesta API per creare una risorsa di indice dei luoghi chiamata *ExamplePlaceIndex* utilizzando il provider di dati *Esri*.

```
POST /places/v0/indexes
Content-type: application/json

{
  "DataSource": "Esri",
  "DataSourceConfiguration": {
    "IntendedUse": "SingleUse"
  },
  "Description": "string",
  "IndexName": "ExamplePlaceIndex",
  "Tags": {
    "Tag1" : "Value1"
  }
}
```

## AWS CLI

Per creare una risorsa relativa all'indice dei luoghi utilizzando i comandi AWS CLI

Utilizza il comando [create-place-index](#).

L'esempio seguente crea una risorsa di indice dei luoghi chiamata *ExamplePlaceIndex* utilizzando *Esri* come fornitore di dati.

```
aws location \  
  create-place-index \  
  --data-source "Esri" \  
  --description "Example place index" \  
  --index-name "ExamplePlaceIndex" \  
  --tags Tag1=Value1
```

### Note

La fatturazione dipende dall'utilizzo. Potresti incorrere in commissioni per l'uso di altri servizi. AWS Per ulteriori informazioni, consulta i [prezzi di Amazon Location Service](#).

## Autenticazione delle tue richieste

Una volta creata una risorsa per l'indice dei luoghi e sei pronto per iniziare a inserire funzionalità di localizzazione nella tua applicazione, scegli come autenticare le tue richieste:

- Per scoprire come accedere ai servizi, consulta [Accedere ad Amazon Location Service](#).
- Se hai un sito Web con utenti anonimi, potresti voler utilizzare le chiavi API o Amazon Cognito.

### Esempio

L'esempio seguente mostra l'utilizzo di una chiave API per l'autorizzazione, l'utilizzo di [AWS JavaScript SDK v3](#) e Amazon Location. [JavaScript Aiutante di autenticazione](#)

```
import { LocationClient, SearchPlaceIndexForTextCommand } from "@aws-sdk/client-location";  
import { withAPIKey } from "@aws/amazon-location-utilities-auth-helper";  
  
const apiKey = "v1.public.your-api-key-value"; // API key  
  
// Create an authentication helper instance using an API key  
const authHelper = await withAPIKey(apiKey);  
  
const client = new LocationClient({  
  region: "<region>", // region containing Cognito pool
```

```
    ..authHelper.getLocationClientConfig(), // Provides configuration required to make
    requests to Amazon Location
  });

  const input = {
    IndexName: "ExamplePlaceIndex",
    Text: "Anyplace",
    BiasPosition: [-123.4567, 45.6789]
  };

  const command = new SearchPlaceIndexForTextCommand(input);

  const response = await client.send(command);
```

## Geocodifica

La geocodifica è un processo che converte il testo, ad esempio un indirizzo, una regione, una ragione sociale o un punto di interesse, in un insieme di coordinate geografiche. È possibile utilizzare le risorse dell'indice di posizione per inviare richieste di geocodifica e incorporare i dati recuperati dalla geocodifica per visualizzare i dati su una mappa per l'applicazione Web o mobile.

Questa sezione illustra come inviare una semplice richiesta di geocodifica e come inviare richieste di geocodifica con specifiche opzionali.

### Geocodifica

È possibile inviare una semplice richiesta di geocodifica utilizzando l'[SearchPlaceIndexForText](#) operazione per convertire un indirizzo in un insieme di coordinate. Una richiesta semplice contiene il seguente parametro obbligatorio:

- **Text**— Un indirizzo, nome, città o regione da convertire in un insieme di coordinate. Ad esempio, la stringa `Any Town`.

Per specificare un numero massimo di risultati per pagina, utilizzate il seguente parametro opzionale:

- **MaxResults**— Limita il numero massimo di risultati restituiti nella risposta alla query.

Puoi utilizzare le AWS CLI o le API di localizzazione di Amazon.



## API

L'esempio seguente è una [SearchPlaceIndexForText](#) richiesta di ricerca nella risorsa dell'indice dei luoghi *ExamplePlaceIndex*, per un indirizzo, nome, città o regione denominata *Any Town*.

```
POST /places/v0/indexes/ExamplePlaceIndex/search/text
Content-type: application/json

{
  "Text": "Any Town",
  "MaxResults": 10
}
```

## AWS CLI

L'esempio seguente è un [search-place-index-for-text](#) comando per cercare nella risorsa dell'indice dei luoghi un indirizzo *ExamplePlaceIndex*, un nome, una città o una regione denominata *Any Town*.

```
aws location \
  search-place-index-for-text \
    --index-name ExamplePlaceIndex \
    --text "Any Town" \
    --max-results 10
```

## Geocodifica vicino a una posizione

Durante la geocodifica, è possibile geocodificare vicino a una determinata posizione con il seguente parametro opzionale:

- **BiasPosition**— La posizione in cui desideri cercare nelle vicinanze. Ciò restringe la ricerca cercando i risultati più vicini alla posizione specificata. Definito come [longitude, latitude]

*L'esempio seguente è una [SearchPlaceIndexForText](#) richiesta di ricerca nella risorsa dell'indice dei luoghi per un indirizzo, un nome, una città o una regione denominata Any Town vicino alla posizione [-123.4567, 45.6789].*

```
POST /places/v0/indexes/ExamplePlaceIndex/search/text
```

```
Content-type: application/json

{
  "Text": "Any Town",
  "BiasPosition": [-123.4567, 45.6789]
}
```

## Geocodifica all'interno di un riquadro di delimitazione

È possibile geocodificare all'interno di un riquadro di delimitazione per restringere i risultati alle coordinate all'interno di un determinato limite utilizzando il seguente parametro opzionale:

- **FilterBoundingBox**— Un riquadro di delimitazione specificato per filtrare i risultati in base alle coordinate all'interno dei confini del riquadro. Definito come [LongitudeSW, LatitudeSW, LongitudeNE, LatitudeNE]

### Note

Una richiesta non può contenere sia **FilterBoundingBox** i **BiasPosition** parametri che. Se si specificano entrambi i parametri nella richiesta, viene restituito un **ValidationException** errore.

L'esempio seguente è una [SearchPlaceIndexForText](#) richiesta di ricerca all'interno di un riquadro di selezione per un indirizzo, un nome, una città o una regione denominata *Any Town*. Il riquadro di delimitazione segue quanto segue:

- *La longitudine dell'angolo sud-ovest è -124,1450.*
- *La latitudine dell'angolo sud-ovest è 41,7045.*
- *La longitudine dell'angolo nord-est è -124,1387.*
- *La latitudine dell'angolo nord-est è 41,7096.*

```
POST /places/v0/indexes/ExamplePlaceIndex/search/text
Content-type: application/json

{
  "Text": "Any Town",
  "FilterBoundingBox": [
```

```
-124.1450,41.7045,  
-124.1387,41.7096  
]  
}
```

## Geocodifica all'interno di un paese

È possibile geocodificare all'interno di uno o più paesi specificati utilizzando il seguente parametro opzionale:

- **FilterCountries**— Il paese o la regione in cui desideri eseguire la geocodifica. È possibile definire fino a 100 paesi in una sola richiesta utilizzando un codice internazionale di tre lettere [ISO 3166](#). Ad esempio, utilizzalo AUS per l'Australia.

L'esempio seguente è una [SearchPlaceIndexForText](#) richiesta di ricerca di un indirizzo, un nome, una città o una regione denominata *Any Town* in Germania e Francia.

```
POST /places/v0/indexes/ExamplePlaceIndex/search/text  
Content-type: application/json  
  
{  
  "Text": "Any Town",  
  "FilterCountries": ["DEU","FRA"]  
}
```

## Filtraggio per categoria

È possibile filtrare le categorie restituite nella richiesta di geocodifica utilizzando il seguente parametro opzionale:

- **FilterCategories**— Le categorie di risultati che desideri vengano restituite nella tua query. È possibile specificare fino a 5 categorie in una richiesta. Puoi trovare l'elenco delle categorie di Amazon Location Service nella sezione [Categorie](#). Ad esempio, puoi specificare di Hotel specificare nella tua query solo gli hotel abituali.

L'esempio seguente è una [SearchPlaceIndexForText](#) richiesta di ricerca di una caffetteria chiamata *Hometown Coffee* negli Stati Uniti.

```
POST /places/v0/indexes/ExamplePlaceIndex/search/text
```

```
Content-type: application/json

{
  "Text": "Hometown Coffee",
  "FilterCategories": ["Coffee Shop"],
  "FilterCountries": ["USA"]
}
```

Per ulteriori dettagli sul filtraggio in base alle categorie, vedere [Posiziona le categorie e filtra i risultati](#)

## Geocodifica in una lingua preferita

È possibile impostare una preferenza di lingua per i risultati della ricerca utilizzando il Language parametro opzionale. Ad esempio, una ricerca di **100 Main St, Anytown, USA** può restituire per 100 Main St, Any Town, USA impostazione predefinita. Ma se si seleziona fr come Language, è possibile che vengano restituiti 100 Rue Principale, Any Town, États-Unis invece i risultati.

- Language— Un codice di lingua da utilizzare per visualizzare i risultati della query. Il valore deve essere un codice di lingua [BCP 47](#) valido. Ad esempio, en per l'inglese.

### Note

Se non Language è specificato o la lingua specificata non è supportata per un risultato, verrà utilizzata la lingua predefinita del partner per quel risultato.

L'esempio seguente è una SearchPlaceIndexforText richiesta di ricerca di un luogo chiamato **Any Town** con la lingua preferita specificata comede.

```
POST /places/v0/indexes/ExamplePlaceIndex/search/text
Content-type: application/json
{
  "Text": "Any Town",
  "Language": "de"
}
```

## Example response

### Example

Di seguito è riportato un esempio di risposta quando chiami l'[SearchPlaceIndexForText](#) operazione dalle API di Amazon Location Places. I risultati includono i [luoghi](#) pertinenti e il [riepilogo](#) della richiesta. Vengono mostrate due risposte, basate sulla selezione di Esri o HERE come partner.

### Example request

```
POST /places/v0/indexes/ExamplePlaceIndex/search/text
Content-type: application/json

{
  "Text": "Amazon",
  "MaxResults": 1,
  "FilterCountries": ["USA"],
  "BiasPosition": [-112.10, 46.32]
}
```

### Example response (Esri)

```
{
  "Results": [
    {
      "Place": {
        "Country": "USA",
        "Geometry": {
          "Point": [
            -112.10667999999998,
            46.3190900000000074
          ]
        },
        "Interpolated": false,
        "Label": "Amazon, MT, USA",
        "Municipality": "Amazon",
        "Region": "Montana",
        "SubRegion": "Jefferson County"
      },
      "Distance": 523.4619749879726,
      "Relevance": 1
    }
  ]
}
```

```
    }
  ],
  "Summary": {
    "BiasPosition": [
      -112.1,
      46.32
    ],
    "DataSource": "Esri",
    "FilterCountries": [
      "USA"
    ],
    "MaxResults": 1,
    "ResultBBox": [
      -112.10667999999998,
      46.319090000000074,
      -112.10667999999998,
      46.319090000000074
    ],
    "Text": "Amazon"
  }
}
```

### Example response (HERE)

```
{
  "Summary": {
    "Text": "Amazon",
    "BiasPosition": [
      -112.1,
      46.32
    ],
    "FilterCountries": [
      "USA"
    ],
    "MaxResults": 1,
    "ResultBBox": [
      -112.10668,
      46.31909,
      -112.10668,
      46.31909
    ],
    "DataSource": "Here"
  },
}
```

```

"Results": [
  {
    "Place": {
      "Label": "Amazon, Jefferson City, MT, United States",
      "Geometry": {
        "Point": [
          -112.10668,
          46.31909
        ]
      },
      "Neighborhood": "Amazon",
      "Municipality": "Jefferson City",
      "SubRegion": "Jefferson",
      "Region": "Montana",
      "Country": "USA",
      "Interpolated": false,
      "TimeZone": {
        "Name": "America/Denver",
        "Offset": -25200
      }
    },
    "PlaceId": "AQAAAIADsn2T3KdrRWeaXLeVEyjNx_JfeTsMB0NVCEAnAZoJ-
o3nqdlJZAdgcT2oWi1w9pS4wXX0k301vsK1GsPyHjV4EJxsu289i3hVO_BUPgP7SFoWAI8BW2v7LvAjQ5NfUPy7a1v9a
et39ZQDWSPLZUzgcjN-6VD2gyKkH0Po7gSm8YSJNSQ",
    "Distance":
523.4619749905755
  }
]
}

```

## Geocodifica inversa

La geocodifica inversa è un processo che converte un insieme di coordinate in testo significativo, ad esempio un indirizzo, una regione, un nome commerciale o un punto di interesse. È possibile utilizzare le risorse dell'indice di posizione per inviare richieste di geocodifica inversa e incorporare i dati recuperati dalla geocodifica inversa per visualizzare i dati su una mappa per l'applicazione Web o mobile.

Questa sezione illustra come inviare una semplice richiesta di geocodifica inversa.

## Geocodifica inversa

È possibile inviare una semplice richiesta per la geocodifica inversa di un insieme di coordinate e convertirle in un indirizzo significativo, un punto di interesse o una posizione generica senza un indirizzo utilizzando l'operazione. [SearchPlaceIndexForPosition](#) Una richiesta semplice contiene il seguente parametro obbligatorio:

- **Position**— Un insieme di coordinate che si desidera convertire in un indirizzo, un punto di interesse o una posizione generale. Definito utilizzando il formato `[longitude, latitude]`.

Per specificare un numero massimo di risultati per pagina, aggiungi il seguente parametro opzionale:

- **MaxResults**— Limita il numero massimo di risultati restituiti nella risposta alla query.

Se si desidera specificare una lingua preferita per i risultati della query, utilizzare il seguente parametro opzionale:

- **Language**— Un codice di lingua da utilizzare per la visualizzazione dei risultati. Il valore deve essere un codice di lingua [BCP 47](#) valido. Ad esempio, en per l'inglese.

### Note

Se non `Language` è specificato o la lingua specificata non è supportata per un risultato, verrà utilizzata la lingua predefinita del partner per quel risultato.

Puoi utilizzare le AWS CLI o le API di localizzazione di Amazon.

### API

*L'esempio seguente è una [SearchPlaceIndexForPosition](#) richiesta di ricerca nella risorsa dell'indice dei luoghi, un indirizzo `ExamplePlaceIndex`, un punto di interesse o una posizione generica vicino alla posizione `[122.3394, 47.6159]`.*

```
POST /places/v0/indexes/ExamplePlaceIndex/search/position
Content-type: application/json
```



```
{
  "Position": [-122.3394,47.6159],
  "MaxResults": 5,
  "Language": "de"
}
```

## AWS CLI

*L'esempio seguente è un [search-place-index-for-position](#) comando per cercare nella risorsa dell'indice dei luoghi un indirizzo significativo `ExamplePlaceIndex`, un punto di interesse o una posizione generale vicino alla posizione [122.3394, 47.6159].*

```
aws location \
  search-place-index-for-position \
    --index-name ExamplePlaceIndex \
    --position -122.3394 47.6159 \
    --max-results 5 \
    --language de
```

## Example response

### Example

Di seguito è riportato un esempio di risposta quando si chiama l'[SearchPlaceIndexForPosition](#) operazione dalle API di Amazon Location Places. I risultati restituiscono i [luoghi](#) pertinenti e il [riepilogo](#) della richiesta. Vengono visualizzate due risposte, basate sulla selezione di Esri o Here come partner.

### Example request

```
POST /places/v0/indexes/ExamplePlaceIndex/search/position
Content-type: application/json

{
  "Position": [-122.3394,47.6159],
  "MaxResults": 1
}
```

## Example response (Esri)

```
{
  "Results": [
    {
      "Place": {
        "AddressNumber": "2111",
        "Country": "USA",
        "Geometry": {
          "Point": [
            -122.33937999999995,
            47.615910000000004
          ]
        },
        "Interpolated": false,
        "Label": "The Spheres, 2111 7th Ave, Seattle, WA, 98121, USA",
        "Municipality": "Seattle",
        "Neighborhood": "Belltown",
        "PostalCode": "98121",
        "Region": "Washington",
        "SubRegion": "King County"
      },
      "Distance": 1.8685861313438727
    }
  ],
  "Summary": {
    "DataSource": "Esri",
    "MaxResults": 1,
    "Position": [
      -122.3394,
      47.6159
    ]
  }
}
```

## Example response (HERE)

```
{
  "Summary": {
    "Position": [
      -122.3394,
      47.6159
    ],
  },
}
```

```

    "MaxResults": 1,
    "DataSource": "Here"
  },
  "Results": [
    {
      "Place": {
        "Label": "2111 7th Ave, Seattle, WA 98121-5114, United States",
        "Geometry": {
          "Point": [
            -122.33938,
            47.61591
          ]
        },
        "AddressNumber": "2111",
        "Street": "7th Ave",
        "Neighborhood": "Belltown",
        "Municipality": "Seattle",
        "SubRegion": "King",
        "Region": "Washington",
        "Country": "USA",
        "PostalCode": "98121-5114",
        "Interpolated": false,
        "TimeZone": {
          "Name": "America/Los_Angeles",
          "Offset": -28800
        }
      },
      "PlaceId": "AQAAAIAADsn2T3KdrRWeaXLeVEyjNx_JfeTsMB0NVCEAnAZoJ-
o3nqd1JZAdgcT2oWi1w9pS4wXX0k301vsK1GsPyHjV4EJxsu289i3hVO_BUPgP7SFoWai8BW2v7LvAjQ5NfUPy7a1v9a
et39ZQDWSPLZUzgcjN-6VD2gyKkH0Po7gSm8YSJNSQ",
      "Distance": 1.868586125090601
    }
  ]
}

```

## Completamento automatico

Il completamento automatico fornisce un feedback tempestivo agli utenti finali durante la digitazione della query di ricerca. Fornisce suggerimenti per indirizzi e punti di interesse sulla base di testo in formato libero parziale o con errori di ortografia. È possibile utilizzare le risorse Place Index

per richiedere suggerimenti di completamento automatico e visualizzare i suggerimenti risultanti nell'applicazione.

Amazon Location non supporta l'archiviazione dei suggerimenti di completamento automatico. Viene restituito un errore se l'indice dei luoghi utilizzato per una chiamata di completamento automatico è configurato per l'uso con i geocodici memorizzati. Per utilizzare i geocodici memorizzati e richiedere suggerimenti, crea e configura indici di luoghi multipli.

Questa sezione descrive come inviare una richiesta di completamento automatico. Inizia con il modulo più semplice della richiesta, quindi mostra i parametri opzionali che è possibile utilizzare per aumentare la pertinenza dei risultati di ricerca con completamento automatico.

## Utilizzo del completamento automatico

È possibile inviare una semplice richiesta di suggerimenti per il completamento automatico utilizzando l'[SearchPlaceIndexForSuggestions](#) operazione. La forma più semplice di richiesta ha un solo parametro obbligatorio, la `queryText`:

- `Text`— Il testo parziale in formato libero da utilizzare per generare suggerimenti di luoghi. Ad esempio, la stringa `eiffel tow`

Per limitare il numero di risultati restituiti, aggiungi il `MaxResults` parametro opzionale:

- `MaxResults`— Limita il numero di risultati restituiti nella risposta alla query.

Puoi utilizzare le API di Amazon Location o il AWS CLI.

### API

L'esempio seguente è una [SearchPlaceIndexForSuggestions](#) richiesta di ricerca nella risorsa dell'indice dei luoghi, *ExamplePlaceIndex*, per un massimo di **5** suggerimenti basati sul nome parziale del luogo *kamp*.

```
POST /places/v0/indexes/ExamplePlaceIndex/search/suggestions
Content-type: application/json

{
  "Text": "kamp",
  "MaxResults": 5
}
```

## AWS CLI

L'esempio seguente è un [search-place-index-for-suggestions](#) comando per cercare nella risorsa dell'indice dei luoghi *ExamplePlaceIndex*, fino a **5** suggerimenti basati sul nome parziale del luogo *kamp*.

```
aws location \
    search-place-index-for-suggestions \
    --index-name ExamplePlaceIndex \
    --text kamp \
    --max-results 5
```

La chiamata a `SearchPlaceIndexForSuggestions` genera un elenco di luoghi con un nome e un ID per ciascuno. Puoi utilizzare questi risultati per presentare suggerimenti su ciò che l'utente potrebbe cercare mentre sta digitando, ad esempio fornendo un elenco a discesa di scelte sotto una casella di testo. *Ad esempio, ecco i risultati dei suggerimenti, basati su un utente che digita kamp.*

```
{
  "Summary": {
    "Text": "kamp",
    "MaxResults": 5,
    "DataSource": "Esri"
  },
  "Results": [
    {
      "Text": "Kampuchea",
      "PlaceId": "AQAAAIADsn2T3KdrRWeaXLeVEyjNx_JfeTsMB0NVCEAnAZoJ-
o3nqd1JZAdgcT2oWi1w9pS4wXX0k301vsK1GsPyHjV4EJxsu289i3hVO_BUPgP7SFoWai8BW2v7LvAjQ5NfUPy7a1v9ajT3
et39ZQDWSPLZUzgcjN-6VD2gyKkH0Po7gSm8YSJNSQ"
    },
    {
      "Text": "Kampoul, Kabul, AFG",
      "PlaceId":
"AQAAAIAAA1mx1_-9ffzXD07rBgo9fh6E01Pd1YKvuT5rz2qBDxqBkhTlgkei0PR2s5sa3YBLxUqQI8bhYmsYcu9R-
DkX3L9QSi3CB5LhNPu160iSFJo6H8S1CrX03QsJALhrr9mdbg0R4R4YDywkHkeBlbnbn7g5C5LI_wYx873WeQZuilwtsGm8j
UeXcb_bg"
    },
    {
      "Text": "Kampala, UGA",
```

```

    "PlaceId":
      "AQAAAIAAzZfZt3qMruKG0byhP6MM0pqy2L8SULLVWT7a3ertLBRS6Q5n7I4s9D7E0nRHADAJ7mL7kvX1Q8HD-
      mpuiATXNJ1Ix4_V_1B15zHe8j1YKMWvXbgb08cMpgR2fqYqZMR1x-
      dfB0080oqujKZldvPIDK1kNe3GwcaqvMWWPMeaGd203brFynubAe-MmFF-Gjz-WBMfUy9og6MV7bkk6NGCA"
    },
    {
      "Text": "Kampar, Riau, IDN",
      "PlaceId": "AQAAAIAAvbXXx-
      sr0i111tH0kPdao0GF7WQ_KaZ444SEnevycp6Gtf_2JWgPfCE5bIQCYwya1uZQpX2a8YJoFm2K7Co14fLu7IK0yYOLhZx4k
    },
    {
      "Text": "Kampung Pasir Gudang Baru, Johor, MYS",
      "PlaceId":
        "AQAAAIAA4HLQHdjUDcaaXLE9wtNIT1cjQYLgkBnMoG2eNN0AaQ8PJoWabLRXmmPUaAj8MAD6vT0i6zqaun5Mixyj7vnYX
    }
  ]
}

```

La sezione successiva spiega come utilizzare i risultati ottenuti PlaceID da questi risultati.

## Utilizzo dei risultati del completamento automatico

La chiamata a `SearchPlaceIndexForSuggestions` genera un elenco di luoghi con un nome e un ID per ciascuno. Puoi utilizzare questi risultati per presentare suggerimenti su ciò che l'utente potrebbe cercare mentre sta digitando, ad esempio fornendo un elenco a discesa di scelte sotto una casella di testo. Quando l'utente sceglie uno dei risultati, puoi quindi richiamare l'[GetPlace](#) operazione con l'ID selezionato per restituire i dettagli di quel luogo, tra cui posizione, indirizzo o altri dettagli.

### Note

A PlaceId è valido solo se tutti i seguenti elementi sono uguali nella richiesta di ricerca originale e nella chiamata a `GetPlace`.

- Cliente Account AWS
- Regione AWS
- Fornitore di dati specificato nella risorsa dell'indice dei luoghi

In genere, lo usi `GetPlace` con le API di localizzazione di Amazon. L'esempio seguente è una [GetPlace](#) richiesta per trovare uno dei suggerimenti della sezione precedente. Questo esempio si basa sul toponimo parziale *kamp*.

```
POST /places/v0/indexes/ExamplePlaceIndex/
places/AQAAAIAADsn2T3KdrRWeaXLeVEyjNx_JfeTsMB0NVCEAnAZoJ-
o3nqdLJZAdgcT2oWi1w9pS4wXX0k301vsKLGsPyHjV4EJxsu289i3hV0_BUPgP7SFoWAI8BW2v7LvAjQ5NfUPy7a1v9ajT3
et39ZQDWSPLZUzgcjN-6VD2gyKkH0Po7gSm8YSJNSQ
```

## Completamento automatico vicino a una posizione

Quando cerchi suggerimenti di luoghi con completamento automatico utilizzando [SearchPlaceIndexForSuggestions](#), puoi ottenere suggerimenti più rilevanti a livello locale aggiungendo il seguente parametro opzionale:

- **BiasPosition**— La posizione in cui desideri cercare nelle vicinanze. Definito come `[longitude, latitude]`.

*L'esempio seguente utilizza una [SearchPlaceIndexForSuggestions](#) richiesta per cercare nella risorsa `ExamplePlaceIndex` dell'indice dei luoghi suggerimenti di luoghi corrispondenti alla query parziale `kamp` vicino alla posizione `[32.5827, 0.3169]`.*

```
POST /places/v0/indexes/ExamplePlaceIndex/search/suggestions
Content-type: application/json

{
  "Text": "kamp",
  "BiasPosition": [32.5827,0.3169]
}
```

*I suggerimenti restituiti per lo stesso `Text` possono essere diversi se ne `BiasPosition` viene scelto uno diverso, ad esempio `[-96.7977, 32.7776]`.*

## Completamento automatico all'interno di un riquadro di delimitazione

Puoi restringere la ricerca con completamento automatico per ricevere solo suggerimenti per i luoghi che si trovano all'interno di un determinato confine aggiungendo il seguente parametro opzionale:

- **FilterBBBox**— Un riquadro di delimitazione specificato per filtrare i risultati in base alle coordinate all'interno dei confini del riquadro. Definito come `[LongitudeSW, LatitudeSW, LongitudeNE, LatitudeNE]`

**Note**

Una richiesta non può contenere sia `FilterBBox` i `BiasPosition` parametri che. Se si specificano entrambi i parametri nella richiesta, viene restituito un `ValidationException` errore.

L'esempio seguente utilizza una [SearchPlaceIndexForSuggestions](#) richiesta per cercare nella risorsa `ExamplePlaceIndex` dell'indice dei luoghi suggerimenti di luoghi corrispondenti alla query parziale `kamp` e che sono contenuti nel riquadro di delimitazione in cui:

- *La longitudine dell'angolo sud-ovest del riquadro di delimitazione è 32,5020.*
- *La latitudine dell'angolo sud-ovest del riquadro di delimitazione è 0,2678.*
- *La longitudine dell'angolo nord-est del riquadro di delimitazione è 32,6129.*
- *La latitudine dell'angolo nord-est del riquadro di delimitazione è 0,3502.*

```
POST /places/v0/indexes/ExamplePlaceIndex/search/suggestions
```

```
Content-type: application/json
```

```
{
  "Text": "kamp",
  "FilterBBox": [
    32.5020, 0.2678,
    32.6129, 0.3502
  ]
}
```

*I suggerimenti restituiti per lo stesso Text sono diversi se ne FilterBBox viene scelto uno diverso, ad esempio [-97.9651, 32.0640, -95.1196, 34.0436].*



## Completamento automatico all'interno di un paese

Puoi restringere la ricerca con completamento automatico per ricevere solo suggerimenti per luoghi che si trovano all'interno di un determinato paese o di un insieme di paesi, aggiungendo il seguente parametro opzionale:

- **FilterCountries**— I paesi in cui desideri cercare suggerimenti di luoghi. Puoi specificare fino a 100 paesi in una richiesta utilizzando un codice internazionale di tre lettere [ISO 3166](#). Ad esempio, utilizzalo AUS per l'Australia.

L'esempio seguente utilizza una [SearchPlaceIndexForSuggestions](#) richiesta per cercare nella risorsa *ExamplePlaceIndex* dell'indice dei luoghi suggerimenti di luoghi corrispondenti alla query parziale *kamp* e contenuti in Uganda, Kenya o Tanzania:

```
POST /places/v0/indexes/ExamplePlaceIndex/search/suggestions
Content-type: application/json

{
  "Text": "kamp",
  "FilterCountries": ["UGA", "KEN", "TZA"]
}
```

*I suggerimenti restituiti per lo stesso Text sono diversi se viene scelto un FilterCountries elenco diverso, ad esempio [» USA «].*

## Example response

*Di seguito è riportato un esempio di risposta dei completamenti automatici suggeriti per l'[SearchPlaceIndexForSuggestions](#) operazione, utilizzando il testo *kamp*.*

```
{
  "Summary": {
    "Text": "kamp",
    "MaxResults": 5,
    "DataSource": "Esri"
  },
  "Results": [
    {
      "Text": "Kampuchea",
```

```

    "PlaceId": "AQAAAIAADsn2T3KdRWeaXLeVEyjNx_JfeTsMB0NVCEAnAZoJ-
o3nqd1JZAdgcT2oWi1w9pS4wXX0k301vsK1GsPyHjV4EJxsu289i3hV0_BUPgP7SFoWai8BW2v7LvAjQ5NfUPy7a1v9ajT3
et39ZQDWSPLZUzgcjN-6VD2gyKkH0Po7gSm8YSJNSQ"
  },
  {
    "Text": "Kampoul, Kabul, AFG",
    "PlaceId":
"AQAAAIAAAA1mx1_-9ffzXD07rBgo9fh6E01Pd1YKvuT5rz2qBDxqBkhTlgkei0PR2s5sa3YBLxUqQI8bhymYcu9R-
DkX3L9QSi3CB5LhNPu160iSFJo6H8S1CrX03QsJALhrr9mdbg0R4R4YDywkHkeB1nbn7g5C5LI_wYx873WeQZuilwtsGm8j
UeXcb_bg"
  },
  {
    "Text": "Kampala, UGA",
    "PlaceId":
"AQAAAIAAzZfZt3qMruKG0byhP6MM0pqy2L8SUL1VWT7a3ertLBRS6Q5n7I4s9D7E0nRHADAj7mL7kvX1Q8HD-
mpuiATXNJ1Ix4_V_1B15zHe8jLYKMWvXbgb08cMpgR2fqYqZMR1x-
dfB0080oqujKZ1dvPIDK1kNe3GwcaqvMWWPMeaGd203brFynubAe-MmFF-Gjz-WBMfUy9og6MV7bkk6NGCA"
  },
  {
    "Text": "Kampar, Riau, IDN",
    "PlaceId": "AQAAAIAAvbXXx-
sr0i111tH0kPdao0GF7WQ_KaZ444SEnevycp6Gtf_2JWgPfCE5bIQCYwya1uZQpX2a8YJoFm2K7Col4fLu7IK0yY0LhZx4k
  },
  {
    "Text": "Kampung Pasir Gudang Baru, Johor, MYS",
    "PlaceId":
"AQAAAIAA4HLQHdjUDcaaXLE9wtNIT1cjQYLgkbnMoG2eNN0AaQ8PJoWabLRXmmPUaAj8MAD6vT0i6zqaun5Mixyj7vnYX
  }
]
}

```

## Utilizzo degli ID dei luoghi

La ricerca di luoghi restituisce un elenco di risultati. La maggior parte dei risultati include un `PlaceId` corrispondente risultato. È possibile utilizzare `PlaceId` a in un' [GetPlace](#) operazione per restituire le informazioni su quel luogo (inclusi nome, indirizzo, posizione o altri dettagli).

**Note**

[SearchPlaceIndexForSuggestions](#)Using restituirà PlaceId risultati per qualsiasi indice di luogo creato con qualsiasi fonte di dati. Usando [SearchPlaceIndexForText](#) o [SearchPlaceIndexForPosition](#) restituirà un PlaceId solo se l'origine dati utilizzata è HERE.

Ciascuno definisce PlaceId in modo univoco il luogo a cui si riferisce, ma un singolo luogo può averne più di uno PlaceId nel tempo e in base al contesto. Le seguenti regole descrivono l'unicità e la longevità di un. PlaceId

- Le chiamate PlaceId restituite che effettui sono specifiche per la tua Account AWS, per la AWS regione e per il fornitore di dati della tua risorsa. PlaceIndex GetPlacetoverà risultati solo quando questi tre attributi corrispondono alla chiamata originale che ha creato ilPlaceId.
- Il PlaceId nome di un luogo cambierà quando cambieranno i dati relativi a quel luogo. Ad esempio, quando l'azienda a cui si riferisce cambia sede o cambia nome.
- Il PlaceId risultato di una chiamata di ricerca ripetuta può cambiare quando il servizio di backend effettua un aggiornamento. La versione precedente PlaceId continuerà a essere trovata, ma le nuove chiamate alla ricerca potrebbero restituire un ID diverso.

PlaceIdÈ una stringa. Non esiste un limite specifico alla lunghezza di aPlaceId. Di seguito è riportato un esempio di validoPlaceId.

```
AQAAAIAADsn2T3KdīRWeaXLeVEyjNx_JfeTsMB0NVCEAnAZoJ-  
o3nqd1JZAdgcT2oWi1w9pS4wXX0k301vsK1GsPyHjV4EJxsu289i3hV0_BUPgP7SFoWAI8BW2v7LvAjQ5NfUPy7a1v9ajT3  
et39ZQDWSPLZUzgcjN-6VD2gyKkH0Po7gSm8YSJNSQ
```

La chiamata GetPlace con a PlaceId per un luogo i cui dati sono stati modificati (ad esempio, una sede aziendale che ha cessato l'attività), genererà un 404 ResourceNotFound errore. Una chiamata GetPlace con un PlaceId messaggio non valido o fuori contesto, ad esempio da un altro Account AWS, restituirà un 400 ValidationException errore.

Sebbene sia possibile utilizzare placelD nelle richieste successive, PlacelD non è destinato a essere un identificatore permanente e l'ID può cambiare tra chiamate API consecutive. Consulta il seguente comportamento di PlacelD per ogni fornitore di dati:

- Esri: gli ID dei luoghi cambieranno almeno ogni trimestre. Il periodo di tempo tipico per queste modifiche sarebbe marzo, giugno, settembre e dicembre. Anche gli ID dei luoghi potrebbero cambiare tra le tipiche variazioni trimestrali, ma ciò sarà molto meno frequente.
- QUI: Ti consigliamo di memorizzare i dati nella cache per non più di una settimana per mantenerli aggiornati. Si può supporre che vengano rilasciati meno dell'1% dei turni ID rispetto al rilascio, ovvero circa 1-2 volte a settimana.
- Grab: I Place ID possono scadere o diventare non validi nelle seguenti situazioni.
  - Operazioni sui dati: il POI può essere rimosso dal database Grab POI da Grab Map Ops sulla base di criteri fondamentali, ad esempio se è chiuso nel mondo reale, viene rilevato come un POI duplicato o contiene informazioni errate. Grab sincronizzerà i dati con l'ambiente Waypoint su base settimanale.
  - POI interpolato: il POI interpolato è un POI temporaneo generato in tempo reale al momento dell'evasione di una richiesta e verrà contrassegnato come derivato nel campo della risposta. `place.result_type` Le informazioni sui POI interpolati verranno conservate per almeno 30 giorni, il che significa che entro 30 giorni potrai ottenere i dettagli dei POI tramite Place ID dall'API Place Details. Dopo 30 giorni, i POI interpolati (sia il Place ID che i dettagli) potrebbero scadere e diventare inaccessibili dall'API Places Details.

## Posiziona le categorie e filtra i risultati

I luoghi sono suddivisi in categorie. Se cerchi un'attività, l'attività potrebbe essere, ad esempio `Restaurant`, una. Anche i risultati di una ricerca di un indirizzo possono essere classificati in base al fatto che corrisponda a un indirizzo, a una via o a un incrocio.

In generale, Amazon Location Service classifica i luoghi in tipi di luoghi. I punti di interesse sono ulteriormente classificati in tipi di punti di interesse.

### Note

Non tutti i risultati avranno delle categorie.

È possibile utilizzare le categorie per filtrare le ricerche di geocodifica.

## Filtrare i risultati

Quando si utilizza `SearchPlaceIndexForText`, è possibile filtrare i risultati restituiti dalle categorie che si desidera utilizzare. Per esempio:

- Se desideri cercare un posto chiamato «Hometown Coffee» e restituire solo risultati classificati come caffetterie, puoi farlo chiamando `SearchPlaceIndexForText` e includendo la categoria `Point of interest Coffee Shop` nel `FilterCategories` parametro.
- Quando cerchi «123 Main St, Anytown, WA, 98123, USA», puoi filtrare i risultati solo in base agli indirizzi, in modo da non ottenere corrispondenze, ad esempio, sul codice postale. Filtra solo gli indirizzi includendo il tipo di luogo, `AddressType` nel `FilterCategories` parametro.

### Note

Non tutti i fornitori di dati supportano il filtraggio o lo supportano allo stesso modo. Per ulteriori informazioni, consulta [Limitazioni di filtraggio per fornitore di dati](#).

La sezione successiva elenca le categorie in base alle quali è possibile filtrare.

## Categories

I seguenti elenchi mostrano le categorie utilizzate da Amazon Location Service per classificare e filtrare. Queste categorie vengono utilizzate in tutte le lingue, indipendentemente dal fatto che il parametro della lingua sia impostato su una lingua diversa.

### Note

Amazon Location Service associa le categorie dei fornitori di dati a questo set di categorie. Se un fornitore di dati inserisce un posto in una categoria che non fa parte dell'elenco delle categorie di Amazon Location Service, la categoria del fornitore verrà inclusa nei risultati come categoria supplementare.

Tipi di luogo: questi tipi vengono utilizzati per indicare il tipo di corrispondenza utilizzata per trovare il risultato.

- `AddressType`— Restituito quando il risultato è stato abbinato a un indirizzo.

- **StreetType**— Restituito quando il risultato è stato abbinato a una strada.
- **IntersectionType**— Restituito quando il risultato è stato abbinato all'intersezione di due strade.
- **PointOfInterestType**— Restituito quando il risultato corrispondeva a un punto di interesse, ad esempio un'attività commerciale o una sede civica.
- **CountryType**— Restituito quando il risultato è stato abbinato a un paese o a una regione principale.
- **RegionType**— Restituito quando il risultato è stato abbinato a una regione all'interno di un paese, ad esempio uno stato o una provincia.
- **SubRegionType**— Restituito quando il risultato è stato abbinato a una sottoregione all'interno di un paese, ad esempio una contea o un'area metropolitana.
- **MunicipalityType**— Restituito quando il risultato è stato abbinato a una città o paese.
- **NeighborhoodType**— Restituito quando il risultato è stato abbinato a un quartiere o a un'area all'interno di una città.
- **PostalCodeType**— Restituito quando il risultato è stato abbinato a un codice postale.

Categorie di punti di interesse: queste categorie vengono utilizzate per indicare il tipo di attività o la sede per i risultati dei punti di interesse.

- Airport
- Amusement Park
- Aquarium
- Art Gallery
- ATM
- Bakery
- Bank
- Bar
- Beauty Salon
- Bus Station
- Car Dealer
- Car Rental
- Car Repair

- Car Wash
- Cemetery
- Cinema
- City Hall
- Clothing Store
- Coffee Shop
- Consumer Electronics Store
- Convenience Store
- Court House
- Dentist
- Embassy
- Fire Station
- Fitness Center
- Gas Station
- Government Office
- Grocery
- Higher Education
- Hospital
- Hotel
- Laundry
- Library
- Liquor Store
- Lodging
- Market
- Medical Clinic
- Motel
- Museum
- Nightlife
- Nursing Home

- Park
- Parking
- Pet Store
- Pharmacy
- Plumbing
- Police Station
- Post Office
- Religious Place
- Restaurant
- School
- Shopping Mall
- Sports Center
- Storage
- Taxi Stand
- Tourist Attraction
- Train Station
- Veterinary Care
- Zoo

## Limitazioni di filtraggio per fornitore di dati

Non tutti i provider dispongono della stessa funzionalità di filtro. La tabella seguente descrive le differenze.

Provider	API con supporto per filtri	Categorie supportate per il filtraggio	Valori restituiti
Esri	<code>SearchPlaceIndexForText</code> , <code>SearchPlaceIndexForSuggestions</code>	Filtra per tipi di luoghi e categorie di punti di interesse.	Le categorie vengono restituite da <code>SearchPlaceIndexForText</code> e <code>SearchPlaceIndexForSuggestions</code>



Provider	API con supporto per filtri	Categorie supportate per il filtraggio	Valori restituiti
			rPosition , e GetPlace
Qui	SearchPlaceIndexForText , SearchPlaceIndexForSuggestions	Filtra solo per tipi di luogo.	Le categorie vengono restituite da SearchPlaceIndexForText e SearchPlaceIndexForSuggestions SearchPlaceIndexForPosition , e GetPlace
Afferra	non supportato	non supportato	non supportato
Open Data (Apri dati)	n/a (i luoghi di ricerca non sono supportati)	N/A	n/a

## Amazon Aurora PostgreSQLfunzioni definite dall'utente per Amazon Location Service

Puoi utilizzare Amazon Location Service per lavorare con coordinate e indirizzi memorizzati nelle tabelle del database per pulire e arricchire i tuoi dati geospaziali.

Per esempio:

- Puoi utilizzare la geocodifica per convertire gli indirizzi in coordinate per normalizzare e colmare le lacune nei dati degli indirizzi memorizzati in una tabella di database.
- È possibile geocodificare gli indirizzi per ottenerne la posizione e utilizzare le coordinate con funzioni spaziali del database, ad esempio una funzione che mostra le righe in un'area specificata.
- È possibile utilizzare dati arricchiti per generare report automatici, ad esempio generare un rapporto automatico che illustra tutti i dispositivi in una determinata area o un rapporto

automatizzato per l'apprendimento automatico che illustra le aree con tassi di errore più elevati durante l'invio di aggiornamenti sulla posizione.

Questo tutorial mostra come formattare e arricchire gli indirizzi memorizzati in una tabella di Amazon Aurora PostgreSQL database utilizzando Amazon Location Service.

- Amazon Aurora PostgreSQL— Un motore di database relazionale completamente gestito, compatibile con MySQL e PostgreSQL, che produce fino a cinque volte il throughput di MySQL e fino a tre volte il throughput di PostgreSQL senza modificare la maggior parte dell'applicazione esistente. Per ulteriori informazioni, consulta [Cos'è Amazon Aurora?](#) nella Guida per l'utente di Amazon Aurora.

#### Important

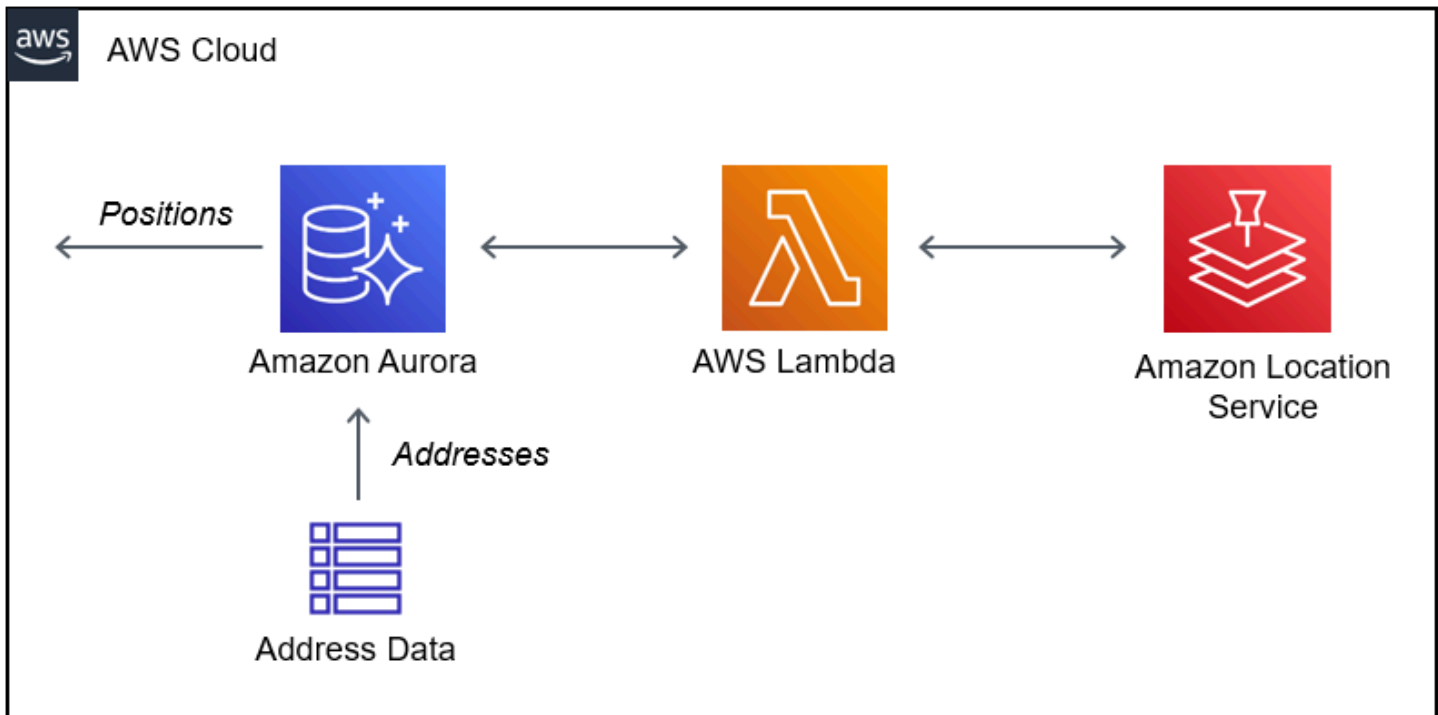
L'applicazione risultante in questo tutorial utilizza un indice dei luoghi che memorizza i risultati della geocodifica. Per informazioni sui costi applicabili per l'archiviazione dei risultati di geocodifica, consulta i prezzi di [Amazon Location Service](#).

Il codice di esempio è disponibile nell'archivio di esempi di Amazon Location Service su [GitHub](#), che include [un AWS CloudFormation modello](#).

#### Argomenti

- [Panoramica](#)
- [Prerequisiti](#)
- [Avvio rapido](#)
- [Crea una risorsa per l'indice dei luoghi](#)
- [Crea una AWS Lambda funzione per la geocodifica](#)
- [Concedi Amazon Aurora PostgreSQL l'accesso a AWS Lambda](#)
- [Richiama la funzione AWS Lambda](#)
- [Arricchimento di un database contenente dati di indirizzo](#)
- [Passaggi successivi](#)

## Panoramica



L'architettura prevede le seguenti integrazioni:

- Questa soluzione utilizza una risorsa Amazon Location Place Index per supportare le query di geocodifica utilizzando l'operazione. `SearchPlaceIndexForText`
- AWS Lambda utilizza un Python Lambda che geocodifica gli indirizzi quando una policy IAM dà il permesso di consentire di AWS Lambda chiamare l'operazione di geocodifica di Amazon Location, `SearchPlaceIndexForText`
- Concedi l'autorizzazione Amazon Aurora PostgreSQL a richiamare la funzione Lambda di geocodifica utilizzando una funzione SQL definita dall'utente.

## Prerequisiti

Prima di iniziare, sono necessari i seguenti prerequisiti:

- Un Amazon Aurora PostgreSQL cluster. Per ulteriori informazioni sulla [creazione di un cluster Amazon Aurora DB](#), consulta la Amazon Aurora User Guide.

**Note**

Se il tuo cluster Amazon Aurora non è disponibile pubblicamente, devi anche configurare Amazon Aurora per la connessione in un cloud AWS Lambda privato virtuale (VPC) nel tuo account. AWS Per ulteriori informazioni, consulta [Concedi Amazon Aurora PostgreSQL l'accesso a AWS Lambda](#).

- Uno strumento di sviluppo SQL per connettersi al cluster. Amazon Aurora PostgreSQL

## Avvio rapido

In alternativa ai passaggi descritti in questo tutorial, puoi avviare uno stack rapido per implementare una AWS Lambda funzione che supporti il funzionamento di Amazon Location.

[SearchPlaceIndexForText](#) Questo configura automaticamente il tuo AWS account per consentire ad Amazon Aurora di effettuare chiamate. AWS Lambda

Una volta configurato il tuo AWS account, dovrai:

- Aggiungi la funzionalità Lambda ad Amazon Aurora. Vedi Aggiungere il ruolo IAM a un cluster Amazon Aurora DB in. [Concedi Amazon Aurora PostgreSQL l'accesso a AWS Lambda](#)
- Carica la funzione definita dall'utente nel tuo database. Per informazioni, consulta [Richiama la funzione AWS Lambda](#).

**Launch Stack**



## Crea una risorsa per l'indice dei luoghi

Inizia creando una risorsa per l'indice dei luoghi per supportare le query di geocodifica.

1. Apri la console Amazon Location Service all'[indirizzo https://console.aws.amazon.com/location/](https://console.aws.amazon.com/location/).
2. Nel riquadro di navigazione a sinistra, scegli Posiziona indici.
3. Compila le seguenti caselle:
  - Nome: inserisci un nome per la risorsa dell'indice dei luoghi. Ad esempio, *AuroraPlaceIndex*. Massimo 100 caratteri. Le voci valide includono caratteri alfanumerici, trattini, punti e caratteri di sottolineatura.

- Descrizione: immettere una descrizione facoltativa. Ad esempio, *Place index per Amazon Aurora*.
4. In Fornitori di dati, scegli un [fornitore di dati](#) disponibile da utilizzare con la tua risorsa place index. Se non hai preferenze, ti consigliamo di iniziare con *Esri*.
  5. In Opzioni di archiviazione dei dati, specificare Sì, i risultati verranno archiviati. Ciò indica che intendi salvare i risultati della geocodifica in un database.
  6. (Facoltativo) In Tag, immettere Key (Chiave) e Value (Valore) di un tag. Questo aggiunge un tag alla nuova risorsa dell'indice dei luoghi. Per ulteriori informazioni, consulta [Tagging delle risorse](#).
  7. Scegli Crea indice dei luoghi.

## Crea una AWS Lambda funzione per la geocodifica

Per creare una connessione tra Amazon Location Service Amazon Aurora PostgreSQL e Amazon, è necessaria una AWS Lambda funzione per gestire le richieste dal motore di database. Questa funzione traduce l'evento della funzione Lambda definita dall'utente e chiama l'operazione Amazon Location. `SearchPlaceIndexForText`

Puoi creare la funzione utilizzando la AWS Lambda console, le o le AWS Command Line Interface API. AWS Lambda

Per creare una funzione Lambda definita dall'utente utilizzando la console

1. Apri la console AWS Lambda all'indirizzo <https://console.aws.amazon.com/lambda/>.
2. Dalla barra di navigazione a sinistra, scegli Funzioni.
3. Scegli Crea funzione e assicurati che sia selezionato Author from zero.
4. Compila le seguenti caselle:
  - Nome della funzione: immettere un nome univoco per la funzione. Le voci valide includono caratteri alfanumerici, trattini e caratteri di sottolineatura senza spazi. *AuroraGeocoderAd* esempio,.
  - Runtime — Scegliete *Python 3.8*.
5. Scegli Crea funzione.
6. Scegli la scheda Codice per aprire l'editor.
7. Sovrascrivi il codice segnato `lambda_function.py` con quanto segue:

```

from os import environ

import boto3
from botocore.config import Config

# load the place index name from the environment, falling back to a default
PLACE_INDEX_NAME = environ.get("PLACE_INDEX_NAME", "AuroraPlaceIndex")

location = boto3.client("location", config=Config(user_agent="Amazon Aurora
  PostgreSQL"))

"""
This Lambda function receives a payload from Amazon Aurora and translates it to
an Amazon Location `SearchPlaceIndex` call and returns the results as-is, to be
post-processed by a PL/pgSQL function.
"""
def lambda_handler(event, context):
    kwargs = {}

    if event.get("biasPosition") is not None:
        kwargs["BiasPosition"] = event["biasPosition"]

    if event.get("filterBBox") is not None:
        kwargs["FilterBBox"] = event["filterBBox"]

    if event.get("filterCountries") is not None:
        kwargs["FilterCountries"] = event["filterCountries"]

    if event.get("maxResults") is not None:
        kwargs["MaxResults"] = event["maxResults"]

    return location.search_place_index_for_text(
        IndexName=PLACE_INDEX_NAME,
        Text=event["text"],
        **kwargs)["Results"]

```

8. Se hai assegnato al tuo indice dei luoghi un nome diverso da quello *AuroraPlaceIndex*, crea una variabile di ambiente denominata `PLACE_INDEX_NAME` per assegnare il nome della risorsa a:
- Dalla scheda Configurazione, scegli Variabili d'ambiente.
  - Scegli Modifica, quindi scegli Aggiungi variabile di ambiente.

- Per chiave: InvioPLACE\_INDEX\_NAME.
  - Per Valore: inserisci il nome della risorsa dell'indice dei luoghi.
9. Scegli Deploy per salvare la funzione aggiornata.
  10. Dal menu a discesa Test, scegli Configura evento di test.
  11. Scegliere Create new test event (Creare nuovo evento di test).
  12. Inserisci il seguente evento di test:

```
{
  "text": "Baker Beach",
  "biasPosition": [-122.483, 37.790],
  "filterCountries": ["USA"]
}
```

13. Scegli Test per testare la funzione Lambda.
14. Scegli la scheda Configurazione.
15. In Configurazione generale: scegli Autorizzazioni.
16. In Ruolo di esecuzione: scegli il nome del ruolo con collegamento ipertestuale per concedere ad Amazon Location Service le autorizzazioni per la tua funzione Lambda.
17. Nella scheda Autorizzazioni: seleziona il menu a discesa Aggiungi autorizzazioni, quindi scegli Crea policy in linea.
18. Seleziona la scheda JSON.
19. Aggiungi la seguente politica IAM:
  - La seguente politica autorizza l'invio della risorsa SearchPlaceIndexForText all'indice dei luoghi *AuroraPlaceIndex*.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "geo:SearchPlaceIndexForText",
      "Resource": "arn:aws:geo:<Region>:<AccountId>:place-index/AuroraPlaceIndex"
    }
  ]
}
```

20. Scegli Esamina la policy.

21. Inserire un nome per la policy. Ad esempio, *AuroraPlaceIndexReadOnly*.
22. Scegli Crea policy.

## Concedi Amazon Aurora PostgreSQL l'accesso a AWS Lambda

Prima di Amazon Aurora PostgreSQL poter richiamare una AWS Lambda funzione, è necessario concedere l'autorizzazione di accesso.

Se il tuo Amazon Aurora PostgreSQL cluster non è accessibile pubblicamente, dovrai prima creare un endpoint VPC per Amazon Aurora per AWS Lambda chiamare la tua funzione Lambda.

Crea un endpoint VPC per AWS Lambda

### Note

Questo passaggio è necessario solo se il Amazon Aurora PostgreSQL cluster non è accessibile pubblicamente.

1. Apri la [Amazon Virtual Private Cloud Console](#).
2. Nella barra di navigazione a sinistra, scegli Endpoints.
3. Seleziona Crea endpoint.
4. Nel filtro Service Name, inserisci «lambda», quindi scegli `com.amazonaws.<region>.lambda`
5. Scegli il VPC contenente il tuo cluster Aurora.
6. Scegli una sottorete per ogni zona di disponibilità.
7. Nel filtro del gruppo di sicurezza, inserisci «default» o il nome del gruppo di sicurezza di cui fa parte il cluster Aurora, quindi scegli il gruppo di sicurezza.
8. Seleziona Crea endpoint.

Crea una policy IAM per concedere l'autorizzazione a richiamare la tua funzione AWS Lambda

1. Apri la [console IAM](#).
2. Nella barra di navigazione a sinistra, espandi Gestione degli accessi per scegliere Policies.
3. Scegli Crea policy.
4. Nella scheda JSON, inserisci la seguente politica:



- Di seguito è riportato un esempio di policy IAM che concede Amazon Aurora PostgreSQL il permesso di richiamare la funzione. AuroraGeocoder AWS Lambda

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "lambda:InvokeFunction",
      "Resource": [
        "arn:aws:lambda:<Region>:<AccountId>:function:AuroraGeocoder"
      ]
    }
  ]
}
```

5. Scegli Avanti: Tag per aggiungere tag opzionali.
6. Seleziona Successivo: Revisione.
7. Rivedi la tua politica e inserisci i seguenti dettagli per la politica:
  - Nome: utilizza caratteri alfanumerici e '+=, .@-\_' . Massimo 128 caratteri.  
*AuroraGeocoderInvoke* Ad esempio,.
  - Descrizione: immettere una descrizione facoltativa. Utilizza caratteri alfanumerici e '+=, .@-\_' . Massimo 1000 caratteri.
8. Scegli Crea policy. Nota l'ARN per questa policy, che usi per associare la policy a un ruolo IAM.

Crea un ruolo IAM per concedere l'autorizzazione ad Amazon Relational Database Service (Amazon RDS)

Creando un ruolo IAM, Amazon Aurora PostgreSQL puoi assumere il ruolo per tuo conto per accedere alla tua funzione Lambda. Per ulteriori informazioni, consulta la pagina relativa alla [creazione di un ruolo per delegare le autorizzazioni a un utente IAM](#) nella Guida per l'utente IAM.

L'esempio seguente è un AWS CLI comando che crea un ruolo denominato

*AuroraGeocoderInvokeRole*:

```
aws iam create-role --role-name rds-lambda-role --assume-role-policy-document '{
  "Version": "2012-10-17",
```

```
"Statement": [  
  {  
    "Effect": "Allow",  
    "Principal": {  
      "Service": "rds.amazonaws.com"  
    },  
    "Action": "sts:AssumeRole"  
  }  
]  
'
```

Collega la tua policy IAM al ruolo IAM

Se hai un ruolo IAM, collega la policy IAM che hai creato.

L'esempio seguente è un AWS CLI comando che collega la policy *AuroraGeocoderInvoke* al ruolo *AuroraGeocoderInvokeRole*.

```
aws iam attach-role-policy --policy-arn AuroraGeocoderInvoke --role-  
name AuroraGeocoderInvokeRole
```

Aggiungi il ruolo IAM a un cluster Amazon Aurora DB

L'esempio seguente è un AWS CLI comando per aggiungere un ruolo IAM a un cluster Amazon Aurora PostgreSQL DB denominato *MyAuroraCluster*.

```
aws rds add-role-to-db-cluster \  
--db-cluster-identifier MyAuroraCluster \  
--feature-name Lambda \  
--role-arn AuroraGeocoderInvokeRole \  
--region your-region
```

## Richiama la funzione AWS Lambda

Dopo aver concesso l'autorizzazione Amazon Aurora PostgreSQL a richiamare la funzione Lambda di geocodifica, puoi creare Amazon Aurora PostgreSQL una funzione definita dall'utente per richiamare la funzione di geocodifica. AWS Lambda Per ulteriori informazioni, consulta [Invocare una AWS Lambda funzione da un cluster Amazon Aurora PostgreSQL DB nella Guida](#) per l'utente di Amazon Aurora.

## Installa le estensioni PostgreSQL richieste

Per installare le estensioni `aws_commons` e le `aws_lambda` estensioni PostgreSQL richieste, consulta [Panoramica sull'uso di una funzione Lambda nella Guida per l'utente di Amazon Aurora](#).

```
CREATE EXTENSION IF NOT EXISTS aws_lambda CASCADE;
```

Installa le estensioni PostGIS richieste

PostGIS è un'estensione di PostgreSQL per l'archiviazione e la gestione delle informazioni spaziali. Per ulteriori informazioni, consulta [Working with the PostGIS nella Amazon Relational Database Service User Guide](#).

```
CREATE EXTENSION IF NOT EXISTS postgis;
```

Creare una funzione SQL definita dall'utente che richiami la funzione Lambda

In un editor SQL, crea una nuova funzione definita dall'utente per richiamare la funzione `f_SearchPlaceIndexForText`: *AuroraGeocoder*

```
CREATE OR REPLACE FUNCTION f_SearchPlaceIndexForText(  
    text text,  
    bias_position geometry(Point, 4326) DEFAULT NULL,  
    filter_bbox box2d DEFAULT NULL,  
    filter_countries text[] DEFAULT NULL,  
    max_results int DEFAULT 1  
)  
RETURNS TABLE (  
    label text,  
    address_number text,  
    street text,  
    municipality text,  
    postal_code text,  
    sub_region text,  
    region text,  
    country text,  
    geom geometry(Point, 4326)  
)  
LANGUAGE plpgsql  
IMMUTABLE  
AS $function$  
begin  
    RETURN QUERY  
    WITH results AS (  

```

```

SELECT json_array_elements(payload) rsp
FROM aws_lambda.invoke(
  aws_commons.create_lambda_function_arn('AuroraGeocoder'),
  json_build_object(
    'text', text,
    'biasPosition',
    CASE WHEN bias_position IS NOT NULL THEN
      array_to_json(ARRAY[ST_X(bias_position), ST_Y(bias_position)])
    END,
    'filterBBox',
    CASE WHEN filter_bbox IS NOT NULL THEN
      array_to_json(ARRAY[ST_XMin(filter_bbox), ST_YMin(filter_bbox),
ST_XMax(filter_bbox), ST_YMax(filter_bbox)])
    END,
    'filterCountries', filter_countries,
    'maxResults', max_results
  )
)
)
)
SELECT
  rsp->'Place'->'Label' AS label,
  rsp->'Place'->'AddressNumber' AS address_number,
  rsp->'Place'->'Street' AS street,
  rsp->'Place'->'Municipality' AS municipality,
  rsp->'Place'->'PostalCode' AS postal_code,
  rsp->'Place'->'SubRegion' AS sub_region,
  rsp->'Place'->'Region' AS region,
  rsp->'Place'->'Country' AS country,
  ST_GeomFromGeoJSON(
    json_build_object(
      'type', 'Point',
      'coordinates', rsp->'Place'->'Geometry'->'Point'
    )
  ) geom
FROM results;
end;
$function$;

```

Chiama la funzione SQL per la geocodifica da Aurora

L'esecuzione dell'istruzione SQL richiama la funzione Lambda *AuroraGeocoder*, che prende i record di indirizzi dalla tabella del database nel Amazon Aurora PostgreSQL database e li geocodifica utilizzando una risorsa di indice dei luoghi.

**Note**

Amazon Aurora PostgreSQL richiama la funzione Lambda per ogni chiamata alla funzione SQL definita dall'utente.

Se si esegue la geocodifica di 50 righe, Amazon Aurora PostgreSQL richiama la funzione Lambda 50 volte. Una chiamata per ogni riga.

La seguente funzione `f_SearchPlaceIndexForText` SQL effettua richieste all'[SearchPlaceIndexForText](#) API di Amazon Location tramite la funzione *AuroraGeocoder* Lambda. La funzione restituisce una `geom` colonna che è una geometria PostGIS, `ST_AsText(geom)` che viene convertita in testo.

```
SELECT *, ST_AsText(geom)
FROM f_SearchPlaceIndexForText('Vancouver, BC');
```

Per impostazione predefinita, il valore restituito conterrà una riga. Per richiedere righe aggiuntive, fino al `MaxResults` limite, esegui la seguente istruzione SQL fornendo un elenco dei risultati (`BiasPosition` limitandolo ai risultati) in Canada.

```
SELECT *
FROM f_SearchPlaceIndexForText('Mount Pleasant', ST_MakePoint(-123.113, 49.260), null,
'{"CAN"}', 5);
```

Per filtrare i risultati utilizzando un riquadro di selezione, passa a come [Box2D](#): `filter_bbox`

- [FilterBBox](#)— Filtra i risultati restituendo le posizioni all'interno di un riquadro di selezione. Si tratta di un parametro facoltativo.

```
SELECT *
FROM f_SearchPlaceIndexForText('Mount Pleasant', null, 'BOX(-139.06 48.30, -114.03
60.00)>:::box2d, '{"CAN"}', 5);
```

[Per ulteriori informazioni sui tipi e le funzioni di PostGIS, consultate PostGIS Reference.](#)

## Arricchimento di un database contenente dati di indirizzo

Puoi creare un indirizzo formattato e contemporaneamente normalizzarlo e geocodificarlo utilizzando l'operazione Amazon Location, `SearchPlaceIndexForText` fornita da una tabella di database con i seguenti dati suddivisi nelle seguenti colonne:

- `id`
- `address`
- `city`
- `state`
- `zip`

```
WITH source_data AS (
  SELECT
    id,
    address || ', ' || city || ', ' || state || ', ' || zip AS formatted_address
  FROM addresses
),
geocoded_data AS (
  SELECT
    *,
    (f_SearchPlaceIndexForText(formatted_address)).*
  FROM source_data
)
SELECT
  id,
  formatted_address,
  label normalized_address,
  ST_Y(geom) latitude,
  ST_X(geom) longitude
FROM geocoded_data
-- limit the number of rows that will be geocoded; remove this to geocode the entire
table
LIMIT 1;
```

L'esempio seguente illustra una riga della tabella databile risultante:

id	formatted_address	normalized_address
latitude	longitude	

```

-----+-----+-----
+-----+-----
42 | 123 Anytown Ave N, Seattle, WA | 123 Anytown Ave N, Seattle, WA, 12345, USA |
47.6223000127926 | -122.336745971039
(1 row)

```

Aggiorna la tabella del database e compila le colonne

L'esempio seguente aggiorna la tabella e popola le colonne con i risultati delle SearchPlaceIndexForText query:

```

WITH source_data AS (
  -- select rows that have not been geocoded and created a formatted address for each
  SELECT
    id,
    address || ', ' || city || ', ' || state || ', ' || zip AS formatted_address
  FROM addresses
  WHERE label IS NULL
  -- limit the number of rows that will be geocoded; remove this to geocode the entire
  table
  LIMIT 1
),
geocoded_data AS (
  -- geocode each row and keep it linked to the source's ID
  SELECT
    id,
    (f_SearchPlaceIndexForText(formatted_address)).*
  FROM source_data
)
UPDATE addresses
-- populate columns
SET
  normalized_address = geocoded_data.label,
  latitude = ST_Y(geocoded_data.geom),
  longitude = ST_X(geocoded_data.geom)
FROM geocoded_data
-- ensure that rows match
WHERE addresses.id = geocoded_data.id;

```

## Passaggi successivi

Il codice di esempio è disponibile nell'archivio di esempi di Amazon Location Service su [GitHub](#), che include [un AWS CloudFormation modello](#).

## Gestione delle risorse relative all'indice dei luoghi

Puoi gestire le risorse dell'indice dei luoghi utilizzando la console Amazon Location o AWS CLI le API di Amazon Location.

### Elenca le tue risorse relative all'indice dei luoghi

Puoi visualizzare l'elenco delle risorse dell'indice dei luoghi utilizzando la console Amazon Location AWS CLI, o le API di Amazon Location:

#### Console

Per visualizzare un elenco di risorse relative all'indice dei luoghi utilizzando la console Amazon Location

1. Apri la console Amazon Location all'[indirizzo https://console.aws.amazon.com/location/](https://console.aws.amazon.com/location/).
2. Scegli Posiziona indici dal riquadro di navigazione a sinistra.
3. Visualizza un elenco delle tue risorse relative all'indice dei luoghi nella sezione Indici I miei luoghi.

#### API

Usa l'[ListPlaceIndexes](#) operazione dalle API di Amazon Location Places.

L'esempio seguente è una richiesta API per ottenere un elenco di risorse dell'indice di posizione nell'AWS account.

```
POST /places/v0/list-indexes
```

Di seguito è riportato un esempio di risposta per [ListPlaceIndexes](#):

```
{
  "Entries": [
    {
      "CreateTime": 2020-10-30T01:38:36Z,
      "DataSource": "Esri",
      "Description": "string",
      "IndexName": "ExamplePlaceIndex",
      "UpdateTime": 2020-10-30T01:40:36Z
    }
  ],
}
```



```
"NextToken": "1234-5678-9012"  
}
```

## CLI

Utilizza il comando [list-place-indexes](#).

L'esempio seguente mostra come AWS CLI ottenere un elenco delle risorse dell'indice dei luoghi nell'AWSaccount.

```
aws location list-place-indexes
```

## Ottieni i dettagli delle risorse dell'indice dei luoghi

Puoi ottenere dettagli su qualsiasi risorsa di indice dei luoghi nel tuo AWS account utilizzando la console Amazon LocationAWS CLI, o le API di Amazon Location:

### Console

Per visualizzare i dettagli di una risorsa relativa all'indice dei luoghi utilizzando la console Amazon Location

1. Apri la console Amazon Location all'[indirizzo https://console.aws.amazon.com/location/](https://console.aws.amazon.com/location/).
2. Scegli Posiziona indici dal riquadro di navigazione a sinistra.
3. In I miei indici dei luoghi, seleziona il link del nome della risorsa dell'indice dei luoghi di destinazione.

### API

Usa l'[DescribePlaceIndex](#)operazione dalle API di Amazon Location Place.

L'esempio seguente è una richiesta API per ottenere i dettagli della risorsa place index per *ExamplePlaceIndex*.

```
GET /places/v0/indexes/ExamplePlaceIndex
```

Di seguito è riportato un esempio di risposta per[DescribePlaceIndex](#):

```
{
```

```
"CreateTime": 2020-10-30T01:38:36Z,  
"DataSource": "Esri",  
"DataSourceConfiguration": {  
  "IntendedUse": "SingleUse"  
},  
"Description": "string",  
"IndexArn": "arn:aws:geo:us-west-2:123456789012:place-indexes/ExamplePlaceIndex",  
"IndexName": "ExamplePlaceIndex",  
"Tags": {  
  "string" : "string"  
},  
"UpdateTime": 2020-10-30T01:40:36Z  
}
```

## CLI

Utilizza il comando [describe-place-index](#).

L'esempio seguente è un AWS CLI modo per ottenere i dettagli della risorsa dell'indice dei luoghi per *ExamplePlaceIndex*.

```
aws location describe-place-index \  
  --index-name "ExamplePlaceIndex"
```

## Eliminare una risorsa relativa all'indice dei luoghi

Puoi eliminare una risorsa di indice dei luoghi dal tuo account AWS utilizzando la console Amazon LocationAWS CLI, o le API di Amazon Location:

### Console

Per eliminare una risorsa relativa all'indice dei luoghi utilizzando la console Amazon Location

#### Warning

Questa operazione elimina la risorsa in modo permanente.

1. Apri la console Amazon Location all'[indirizzo https://console.aws.amazon.com/location/](https://console.aws.amazon.com/location/).
2. Scegli Posiziona indici dal riquadro di navigazione a sinistra.

3. In My place index, seleziona la risorsa dell'indice dei luoghi di destinazione.
4. Scegliete Elimina l'indice dei luoghi.

## API

Usa l'[DeletePlaceIndex](#) operazione dalle API di Amazon Location Places.

L'esempio seguente è una richiesta API per eliminare la risorsa *ExamplePlaceIndex* place index.

```
DELETE /places/v0/indexes/ExamplePlaceIndex
```

Di seguito è riportato un esempio di risposta riuscita per [DeletePlaceIndex](#):

```
HTTP/1.1 200
```

## CLI

Utilizza il comando [delete-place-index](#).

L'esempio seguente è un AWS CLI comando per eliminare la risorsa place index *ExamplePlaceIndex*.

```
aws location delete-place-index \  
  --index-name "ExamplePlaceIndex"
```

## Calcolo dei percorsi utilizzando Amazon Location Service

Amazon Location ti consente di selezionare un fornitore di dati per il calcolo di un percorso creando e configurando una risorsa per il calcolo del percorso.

Puoi utilizzare la risorsa route calculator per [calcolare un percorso](#) in base a parametri specifici utilizzando l'AWSSDK o gli endpoint dell'API REST. Usa questa risorsa di calcolo delle rotte per calcolare i percorsi tra un'origine, una destinazione e fino a 23 waypoint per diverse modalità di trasporto, evitamenti e condizioni del traffico.

[Puoi anche utilizzare la risorsa di calcolo del percorso per creare input per i tuoi algoritmi o prodotti di pianificazione del percorso calcolando una matrice di percorsi.](#) Calcola il tempo di viaggio e la distanza da percorrere tra una serie di posizioni di partenza e una serie di posizioni di destinazione. Il

software di pianificazione del percorso può utilizzare i dati relativi a tempo e distanza per ottimizzare un percorso o una serie di percorsi, ad esempio se stai pianificando più percorsi di consegna e desideri trovare il percorso e l'orario migliori per ogni fermata. È possibile calcolare una matrice di percorsi per diverse modalità di trasporto, evitamenti e condizioni del traffico.

#### Note

Per una panoramica dei concetti di routing, vedere. [Route](#)

## Argomenti

- [Prerequisiti](#)
- [Calcola un percorso](#)
- [Pianificazione del percorso con una matrice di percorso](#)
- [Posizioni non situate su una strada](#)
- [Ora di partenza](#)
- [Modalità di viaggio](#)
- [Gestione delle risorse del calcolatore di percorso](#)

## Prerequisiti

Prima di iniziare a calcolare i percorsi, segui i passaggi preliminari:

### Argomenti

- [Crea una risorsa per il calcolo dei percorsi](#)
- [Autenticazione delle tue richieste](#)

## Crea una risorsa per il calcolo dei percorsi

Prima di poter calcolare un percorso, crea una risorsa per il calcolo del percorso nel tuo AWS account.

Quando crei una risorsa per il calcolo del percorso, puoi scegliere tra i fornitori di dati disponibili:

1. Esri: per ulteriori informazioni sulla copertura di Esri nella regione di interesse, consulta i [dettagli di Esri sulle reti stradali e sulla](#) copertura del traffico.

2. Tecnologie HERE — Per ulteriori informazioni sulla copertura di HERE nella tua area di interesse, consulta [HERE Car Routing Coverage](#) e [HERE Truck Routing coverage](#).
3. Grab — Per ulteriori informazioni sulla copertura di Grab, consulta [Paesi/regioni e aree coperte](#)

### Note

Se la tua applicazione esegue il tracciamento o l'instradamento di risorse utilizzate nella tua azienda, ad esempio veicoli per la consegna o dipendenti, non devi utilizzare Esri come provider di geolocalizzazione. Per ulteriori dettagli, consulta la sezione 82 dei [Termini di servizio di AWS](#).

Puoi farlo utilizzando la console Amazon Location ServiceAWS CLI, o le API Amazon Location.

## Console

Per creare una risorsa per il calcolo del percorso utilizzando la console Amazon Location

1. Apri la console Amazon Location all'[indirizzo https://console.aws.amazon.com/location/](https://console.aws.amazon.com/location/).
2. Nel riquadro di navigazione a sinistra, scegli Calcolatori di percorso.
3. Scegli Crea calcolatore di percorso.
4. Compila le seguenti caselle:
  - Nome: inserisci un nome per la risorsa del calcolatore del percorso. Ad esempio, *ExampleCalculator*. Massimo 100 caratteri. Le voci valide includono caratteri alfanumerici, trattini, punti e caratteri di sottolineatura.
  - Descrizione: immettere una descrizione opzionale.
5. Per i fornitori di dati, scegli un [fornitore di dati](#) da utilizzare come calcolatore di percorso.
6. (Facoltativo) In Tag, immettere Key (Chiave) e Value (Valore) di un tag. Questo aggiunge un tag alla tua nuova risorsa per il calcolo del percorso. Per ulteriori informazioni, consulta [Tagging delle risorse](#).
7. Scegli Crea calcolatore di percorso.

## API

Per creare una risorsa per il calcolo del percorso utilizzando le API di Amazon Location

Usa l'[CreateRouteCalculator](#) operazione dalle API di Amazon Location Places.

*L'esempio seguente è una richiesta API per creare una risorsa di calcolo del percorso chiamata `ExampleCalculator` utilizzando il provider di dati `Esri`.*

```
POST /routes/v0/calculators
Content-type: application/json

{
  "CalculatorName": "ExampleCalculator",
  "DataSource": "Esri",
  "Description": "string",
  "Tags": {
    "Tag1" : "Value1"
  }
}
```

## AWS CLI

Per creare una risorsa per il calcolo del percorso utilizzando i comandi AWS CLI

Utilizza il comando `create-route-calculator`.

L'esempio seguente crea una risorsa per il calcolo del percorso chiamata *ExampleCalculator* utilizzando *Esri come fornitore* di dati.

```
aws location \
  create-route-calculator \
  --calculator-name "ExampleCalculator" \
  --data-source "Esri" \
  --tags Tag1=Value1
```

### Note

La fatturazione dipende dall'utilizzo. Potresti incorrere in commissioni per l'uso di altri servizi. AWS Per ulteriori informazioni, consulta i [prezzi di Amazon Location Service](#).

## Autenticazione delle tue richieste

Una volta creata una risorsa per il calcolo del percorso e sei pronto per iniziare a integrare le funzionalità di localizzazione nella tua applicazione, scegli come autenticare le tue richieste:

- Per scoprire come accedere ai servizi, consulta [Accedere ad Amazon Location Service](#).
- Se hai un sito Web con utenti anonimi, potresti voler utilizzare le chiavi API o Amazon Cognito.

### Esempio

L'esempio seguente mostra l'utilizzo di una chiave API per l'autorizzazione, l'utilizzo di [AWS JavaScript SDK v3](#) e Amazon Location. [JavaScript Aiutante di autenticazione](#)

```
import { LocationClient, CalculateRouteCommand } from "@aws-sdk/client-location";
import { withAPIKey } from "@aws/amazon-location-utilities-auth-helper";

const apiKey = "v1.public.your-api-key-value"; // API key

// Create an authentication helper instance using an API key
const authHelper = await withAPIKey(apiKey);

const client = new LocationClient({
  region: "<region>", // region containing Cognito pool
  ...authHelper.getLocationClientConfig(), // Provides configuration required to make
  requests to Amazon Location
});

const input = {
  CalculatorName: "ExampleCalculator",
  DeparturePosition: [-123.4567, 45.6789],
  DestinationPosition: [-123.123, 45.234],
};

const command = new CalculateRouteCommand(input);

const response = await client.send(command);
```

## Calcola un percorso

Puoi utilizzare Amazon Location Service per calcolare i percorsi tra un'origine e una destinazione, con un massimo di 23 waypoint lungo il percorso, per diverse modalità di trasporto, evitamenti e condizioni del traffico.

### Note

Devi prima creare una risorsa per il calcolo del percorso e configurare l'autenticazione per le tue richieste ad Amazon Location. Per ulteriori informazioni, consulta [Prerequisiti](#).

## Inizia a calcolare i percorsi

Invia una richiesta semplice utilizzando l'[CalculateRoute](#) operazione. Una richiesta semplice contiene i seguenti campi obbligatori:

- **DeparturePosition**— La posizione di partenza da cui calcolare il percorso. Definita come [longitude, latitude]
- **DestinationPosition**— La posizione finale verso la quale calcolare il percorso. Definito come [longitude, latitude].

### Note

Se specifichi una posizione di partenza o di destinazione che non si trova su una strada, Amazon Location [sposta la posizione sulla strada più vicina](#).

Facoltativamente, puoi specificare [waypoint](#), un [orario di partenza](#) e una [modalità di viaggio](#) nella tua richiesta.

Puoi utilizzare le AWS CLI o le API di localizzazione di Amazon.

## API

L'esempio seguente è una CalculateRoute richiesta che utilizza la risorsa route calculator. *ExampleCalculator* La richiesta specifica il calcolo di un percorso da una posizione di partenza [-122.7565, 49.0021] a una posizione di destinazione [-122.3394, 47.6159].



```
POST /routes/v0/calculators/ExampleCalculator/calculate/route
Content-type: application/json
{
  "DeparturePosition": [-122.7565,49.0021],
  "DestinationPosition": [-122.3394, 47.6159]
}
```

## AWS CLI

L'esempio seguente è un comando che utilizza la risorsa del calcolatore di rotta. `calculate-route` *ExampleCalculator* La richiesta specifica il calcolo di un percorso da una posizione di partenza [-122.7565, 49.0021] a una posizione di destinazione [-122.3394, 47.6159].

```
aws location \
  calculate-route \
    --calculator-name ExampleCalculator \
    --departure-position -122.7565 49.0021 \
    --destination-position -122.3394 47.6159
```

Per impostazione Distance predefinita, la risposta viene restituita in chilometri. È possibile modificare l'unità di misura in miglia utilizzando il seguente parametro opzionale:

- `DistanceUnit`— specifica il sistema di unità da utilizzare per i risultati della distanza.

## Example

```
POST /routes/v0/calculators/ExampleCalculator/calculate/route
Content-type: application/json
{
  "DeparturePosition": [-122.7565,49.0021],
  "DestinationPosition": [-122.3394, 47.6159],
  "DistanceUnit": "Miles"
}
```

## Impostazione dei waypoint

Quando si calcola un percorso, è possibile specificare fino a 23 punti di scalo intermedi tra la posizione di partenza e la posizione di destinazione utilizzando le posizioni dei waypoint.

- **WaypointPositions**— Specifica un elenco ordinato di posizioni intermedie da includere lungo un percorso tra la posizione di partenza e la posizione di destinazione.

#### Note

Se specifichi una posizione del waypoint che non si trova su una strada, Amazon Location sposta la posizione sulla strada più vicina.

## Example

La seguente [CalculateRoute](#) richiesta calcola un percorso con 2 waypoint:

- La posizione di partenza è [-122.7565, 49.0021] e la posizione di destinazione è [-122.3394, 47.6159].
- **WaypointPositions** Per il parametro di richiesta:
  - La prima posizione di sosta è [-122.1884, 48.0936].
  - *La seconda posizione di stop over è [-122.3493, 47.6205].*
- *Per includere la geometria del leg linestring tra questi due waypoint, imposta il seguente parametro opzionale su true:*
  - **IncludeLegGeometry**— Include la geometria di ogni percorso tra un paio di posizioni nella risposta.

```
POST /routes/v0/calculators/ExampleCalculator/calculate/route
Content-type: application/json
{
  "DeparturePosition": [-122.7565,49.0021],
  "DestinationPosition": [-122.3394, 47.6159],
  "WaypointPositions":[
    [-122.1884,48.0936],
    [-122.3493,47.6205]
  ],
  "IncludeLegGeometry": true
}
```

## Example response

Di seguito è riportato un esempio di richiesta con la risposta corrispondente quando si chiama l'[CalculateRoute](#) operazione dall'API Amazon Location Routes con il valore `IncludeLegGeometry` impostato su `true`, che include la geometria linestring di ogni percorso tra un paio di posizioni nella risposta.

## Example request

```
POST /routes/v0/calculators/ExampleCalculator/calculate/route
Content-type: application/json
{
  "DeparturePosition": [-122.7565,49.0021],
  "DestinationPosition": [-122.3394, 47.6159],
  "IncludeLegGeometry": true
}
```

## Example response

```
{
  "Legs": [
    {
      "Distance": 178.5,
      "DurationSeconds": 6480,
      "EndPosition": [-122.3394,47.6159],
      "Geometry": {
        "LineString": [
          [-122.7565,49.0021],
          [-122.3394,47.6159]
        ]
      },
      "StartPosition": [-122.7565,49.0021],
      "Steps": [
        {
          "Distance": 178.5,
          "DurationSeconds": 6480,
          "EndPosition": [-122.3394,47.6159],
          "GeometryOffset": 0,
          "StartPosition": [-122.7565,49.0021]
        }
      ]
    }
  ],
}
```

```
"Summary": {
  "DataSource": "Esri",
  "Distance": 178.5,
  "DistanceUnit": "Kilometers",
  "DurationSeconds": 6480,
  "RouteBBox": [
    -122.7565,49.0021,
    -122.3394,47.6159
  ]
}
```

## Pianificazione del percorso con una matrice di percorso

Puoi utilizzare Amazon Location Service per creare input per il tuo software di pianificazione e ottimizzazione dei percorsi. Puoi creare risultati di percorso, inclusi tempo di viaggio e distanza da percorrere, per percorsi tra una serie di posizioni di partenza e una serie di posizioni di destinazione.

Ad esempio, date le posizioni di partenza A e B e le posizioni di destinazione X e Y, Amazon Location Service restituirà il tempo di viaggio e la distanza percorsa per i percorsi da A a X, da A a Y, da B a X e da B a Y.

Puoi calcolare i percorsi con diverse modalità di trasporto, evitamenti e condizioni del traffico. Ad esempio, è possibile specificare che il veicolo è un camion lungo 35 piedi e il percorso calcolato utilizzerà tali restrizioni per determinare il tempo di viaggio e la distanza da percorrere.

Il numero di risultati restituiti (e percorsi calcolati) è il numero di posizioni di partenza moltiplicato per il numero di posizioni di destinazione. Ti viene addebitato ogni itinerario calcolato, non ogni richiesta al servizio, quindi una matrice di itinerari con 10 partenze e 10 destinazioni verrà fatturata come 100 rotte.

### Calcolo di una matrice di itinerari

È possibile calcolare una matrice di percorsi tra un insieme di posizioni di partenza e un insieme di posizioni di destinazione. I risultati del percorso includeranno il tempo di viaggio e la distanza da percorrere.

#### Prerequisito

- Devi prima creare una risorsa per il calcolo del percorso e configurare l'autenticazione per le tue richieste ad Amazon Location. Per ulteriori informazioni, consulta [Prerequisiti](#).

Invia una richiesta utilizzando l'[CalculateRouteMatrix](#) operazione. Una richiesta minima contiene i seguenti campi obbligatori:

- **DeparturePositions**— L'insieme delle posizioni di partenza per le quali calcolare i percorsi. Definito come una matrice di [longitude, latitude]
- **DestinationPositions**— L'insieme delle posizioni finali per le quali calcolare i percorsi. Definito come una matrice di [longitude, latitude].

#### Note

Se specifichi una posizione di partenza o di destinazione che non si trova su una strada, Amazon Location [sposta la posizione sulla strada più vicina](#).

Facoltativamente, puoi specificare un [orario di partenza](#) e una [modalità di viaggio](#) nella tua richiesta.

Puoi utilizzare le AWS CLI o le API di localizzazione di Amazon.

## API

L'esempio seguente è una `CalculateRouteMatrix` richiesta che utilizza la risorsa `route calculator`. *ExampleCalculator* La richiesta specifica il calcolo della matrice dei percorsi dalle posizioni di partenza `[-122.7565, 49.0021]` e `[-122.2014, 47.6101]` alle posizioni di destinazione `[-122.3394, 47.6159]` e `[-122.4813, 48.7511]`.

```
POST /routes/v0/calculators/ExampleCalculator/calculate/route-matrix
Content-type: application/json
{
  "DeparturePositions": [
    [-122.7565, 49.0021],
    [-122.2014, 47.6101]
  ],
  "DestinationPositions": [
    [-122.3394, 47.6159],
    [-122.4813, 48.7511]
  ]
}
```

## AWS CLI

L'esempio seguente è un comando *ExampleCalculator* che utilizza la risorsa del calcolatore di rotta. *calculate-route-matrix* La richiesta specifica il calcolo della matrice dei percorsi dalle posizioni di partenza *[-122.7565, 49.0021]* e *[-122.2014, 47.6101]* alle posizioni di destinazione *[-122.3394, 47.6159]* e *[-122.4813, 48.7511]*.

```
aws location \  
  calculate-route-matrix \  
    --calculator-name ExampleCalculator \  
    --departure-positions "[-122.7565, 49.0021],[-122.2014, 47.6101]" \  
    --destination-positions "[-122.3394, 47.6159],[-122.4813, 48.7511]"
```

Per impostazione predefinita, la risposta viene restituita in chilometri. Distance È possibile modificare l'unità di misura in miglia utilizzando il seguente parametro opzionale:

- **DistanceUnit**— specifica il sistema di unità da utilizzare per i risultati della distanza.

## Example

```
POST /routes/v0/calculators/ExampleCalculator/calculate/route-matrix  
Content-type: application/json  
{  
  "DeparturePositions": [  
    [-122.7565, 49.0021],  
    [-122.2014, 47.6101]  
  ],  
  "DestinationPositions": [  
    [-122.3394, 47.6159],  
    [-122.4813, 48.7511]  
  ],  
  "DistanceUnit": "Miles"  
}
```

## Restrizioni sulle posizioni di partenza e di destinazione

Quando si calcola una matrice di percorso, esistono restrizioni sulle posizioni di partenza e di destinazione. Queste restrizioni variano a seconda del provider utilizzato dalla *RouteCalculator* risorsa.

Limitazione	Esri	Afferra	QUI
Numero di posizioni	Fino a 10 posizioni di partenza e 10 posizioni di destinazione.	Fino a 350 posizioni di partenza e 350 posizioni di destinazione.	Fino a 350 posizioni di partenza e 350 posizioni di destinazione.  Per i percorsi più lunghi, si applicano restrizioni aggiuntive. Consulta la <a href="#">sezione</a> .
Distanza tra le posizioni	Qualsiasi coppia di posizioni di partenza e di destinazione deve trovarsi entro 400 km l'una dall'altra (40 km per i percorsi pedonali).		Tutte le posizioni di partenza e di destinazione devono rientrare in un cerchio di 180 km di diametro.  Per i percorsi più lunghi, si applicano restrizioni aggiuntive. Consulta la <a href="#">sezione</a> .
Lunghezza del percorso	I percorsi non verranno completati se il tempo di percorrenza totale per il percorso è superiore a 400 minuti.		I percorsi che si discostano per più di 10 km dall'esterno di un cerchio attorno ai punti di partenza e di destinazione non verranno calcolati.  Per i percorsi più lunghi, si applicano restrizioni aggiuntive. Consulta la <a href="#">sezione</a> .
Regioni	Il calcolo di una matrice di percorso	Disponibile nel sud-est asiatico. Per un	Nessuna restrizione aggiuntiva.

Limitazione	Esri	Afferra	QUI
	non è supportato in Corea.	elenco dei paesi/aree geografiche supportate e ulteriori informazioni, consulta. <a href="#">Paesi/regioni e aree coperte</a>	

## Pianificazione di percorsi più lunghi

Il calcolo di una matrice dei risultati del percorso è utile per una pianificazione efficiente del percorso, ma il calcolo può richiedere del tempo. Tutti i fornitori di dati di Amazon Location Service impongono limitazioni al numero di percorsi o alla distanza dei percorsi che possono essere calcolati. Ad esempio, HERE consente di creare percorsi tra 350 posizioni di partenza e di destinazione, ma tali posizioni devono rientrare in un cerchio di 180 km. Cosa succede se desideri pianificare percorsi più lunghi?

Puoi calcolare una matrice di percorsi con lunghezze illimitate per un numero inferiore di percorsi utilizzando, a RouteCalculator con HERE come fornitore di dati. Ciò non cambia il modo in cui chiami l'[CalculateRouteMatrix](#) API, Amazon Location consente semplicemente percorsi più lunghi quando soddisfatti i requisiti.

I requisiti per il calcolo di percorsi di maggiore lunghezza sono:

- È RouteCalculator necessario utilizzare il fornitore di dati HERE.
- Il numero di posizioni di partenza non deve essere superiore a 15.
- Il numero totale di rotte da calcolare non deve essere superiore a 100.
- L'itinerario a lunga distanza non è consentito per gli itinerari su camion che evitano i pedaggi quando i percorsi superano i 1.000 km. Questa combinazione è più lenta da calcolare e può causare il timeout della chiamata. È possibile calcolare questi percorsi singolarmente con l'[CalculateRoute](#) operazione.

Se la chiamata non soddisfa questi requisiti (ad esempio, si richiedono 150 calcoli di percorso in una singola chiamata), CalculateRouteMatrix tornerà a consentire solo le regole di routing più brevi. È quindi possibile calcolare i percorsi, purché le posizioni si trovino all'interno di un cerchio di 180 km.

Nel calcolare percorsi più lunghi, tieni a mente questi punti:



- I percorsi più lunghi possono richiedere più tempo per essere calcolati, anche più del tempo massimo per le API di localizzazione di Amazon. Se ricevi timeout frequenti con percorsi specifici, puoi provare un numero inferiore di percorsi per ogni chiamata verso `CalculateRouteMatrix`.
- Se aggiungi più destinazioni o posizioni di partenza alla tua `CalculateRouteMatrix` richiesta, l'operazione può passare alla modalità più limitata e potresti ricevere un errore relativo a un percorso che può essere calcolato senza problemi quando ci sono meno percorsi da creare. In questo caso, riduci il numero di posizioni di destinazione o di partenza ed effettua più richieste per ottenere il set completo di calcoli del percorso di cui hai bisogno.

## Example response

Di seguito è riportato un esempio di richiesta con la risposta corrispondente quando si chiama l'[CalculateRouteMatrix](#) operazione dall'API Amazon Location Routes.

### Example request

```
POST /routes/v0/calculators/ExampleCalculator/calculate/route-matrix
Content-type: application/json
{
  "DeparturePositions": [
    [-122.7565,49.0021],
    [-122.2014,47.6101]
  ],
  "DestinationPositions": [
    [-122.3394, 47.6159],
    [-122.4813,48.7511]
  ]
}
```

### Example response

```
{
  "RouteMatrix": [
    [
      {
        "Distance": 178.764,
        "DurationSeconds": 7565
      },
      {
        "Distance": 39.795,
        "DurationSeconds": 1955
      }
    ]
  ]
}
```

```

    }
  ],
  [
    {
      "Distance": 15.31,
      "DurationSeconds": 1217
    },
    {
      "Distance": 142.506,
      "DurationSeconds": 6279
    }
  ]
],
"Summary": {
  "DataSource": "Here",
  "RouteCount": 4,
  "ErrorCount": 0,
  "DistanceUnit": "Kilometers"
}
}

```

## Posizioni non situate su una strada

Quando usi `CalculateRoute` o `CalculateRouteMatrix`, se specifichi una posizione di partenza, destinazione o waypoint che non si trova su una strada, Amazon Location sposta la posizione su una strada vicina.

La seguente [CalculateRoute](#) richiesta specifica una posizione di partenza e una posizione di destinazione che non si trovano su una strada:

```

POST /routes/v0/calculators/ExampleCalculator/calculate/route
Content-type: application/json
{
  "DeparturePosition": [-123.128014, 49.298472],
  "DestinationPosition": [-123.134701, 49.294315]
}

```

La risposta risultante restituisce una posizione che corrisponde a una strada vicina:

```

{
  "Legs": [

```

```
{
  "StartPosition": [-123.12815, 49.29717],
  "EndPosition": [-123.13375, 49.2926],
  "Distance": 4.223,
  "DurationSeconds": 697,
  "Steps": [
    {
      "StartPosition": [ -123.12815, 49.29717 ],
      "EndPosition": [ -123.12806, 49.29707 ],
      "Distance": 0.013,
      "DurationSeconds": 8
    },
    {
      "StartPosition": [ -123.12806, 49.29707 ],
      "EndPosition": [ -123.1288, 49.29659 ],
      "Distance": 0.082,
      "DurationSeconds": 36
    },
    {
      "StartPosition": [ -123.1288, 49.29659 ],
      "EndPosition": [ -123.12021, 49.29853 ],
      "Distance": 0.742,
      "DurationSeconds": 128
    },
    {
      "StartPosition": [ -123.12021, 49.29853 ],
      "EndPosition": [ -123.1201, 49.29959 ],
      "Distance": 0.131,
      "DurationSeconds": 26
    },
    {
      "StartPosition": [ -123.1201, 49.29959 ],
      "EndPosition": [ -123.13562, 49.30681 ],
      "Distance": 1.47,
      "DurationSeconds": 238
    },
    {
      "StartPosition": [ -123.13562, 49.30681 ],
      "EndPosition": [ -123.13693, 49.30615 ],
      "Distance": 0.121,
      "DurationSeconds": 28
    },
    {
      "StartPosition": [ -123.13693, 49.30615 ],
```

```

        "EndPosition": [ -123.13598, 49.29755 ],
        "Distance": 0.97,
        "DurationSeconds": 156
    },
    {
        "StartPosition": [ -123.13598, 49.29755 ],
        "EndPosition": [ -123.13688, 49.29717 ],
        "Distance": 0.085,
        "DurationSeconds": 15
    },
    {
        "StartPosition": [ -123.13688, 49.29717 ],
        "EndPosition": [ -123.13375, 49.2926 ],
        "Distance": 0.609,
        "DurationSeconds": 62
    }
]
}
],
"Summary": {
    "RouteBBox": [ -123.13693, 49.2926, -123.1201, 49.30681 ],
    "DataSource": "Here",
    "Distance": 4.223,
    "DurationSeconds": 697,
    "DistanceUnit": "Kilometers"
}
}

```

## Ora di partenza

Per impostazione predefinita, quando chiami `CalculateRoute` o `CalculateRouteMatrix`, se non fornisci un orario di partenza nella richiesta, i percorsi calcolati riflettono le condizioni di traffico ottimali.

Puoi impostare un orario di partenza specifico per utilizzare le condizioni di traffico in tempo reale e predittive del fornitore di dati scelto, utilizzando una delle seguenti opzioni:

- `DepartNow`— Se impostato su *true*, utilizza le condizioni del traffico in tempo reale per calcolare il percorso di viaggio più veloce.
- `DepartureTime`— Se fornito, utilizza condizioni di traffico predittive e note per l'orario richiesto. Definito nel seguente [formato](#): `YYYY-MM-DDThh:mm:ss.sssZ`.

## Example

La seguente [CalculateRoute](#) richiesta fissa l'orario di partenza al 2 luglio 2024 alle 12:15:20 UTC.

```
POST /routes/v0/calculators/ExampleCalculator/calculate/route
Content-type: application/json
{
  "DeparturePosition": [-122.7565,49.0021],
  "DestinationPosition": [-122.3394, 47.6159],
  "WaypointPositions":[
    [-122.1884,48.0936],
    [-122.3493,47.6205]
  ]
  "IncludeLegGeometry": true,
  "DepartureTime": 2024-07-02T12:15:20.000Z,
}
```

## Modalità di viaggio

Puoi impostare una modalità di viaggio quando usi `CalculateRoute` o `CalculateRouteMatrix`. La modalità di marcia influisce sulla velocità di marcia e sulla compatibilità stradale. Sebbene la modalità di viaggio predefinita sia l'auto, puoi specificare quale modalità di viaggio utilizzare mentre percorri un itinerario con il seguente parametro opzionale:

- `TravelMode`— specifica la modalità di trasporto durante il calcolo di un percorso, ad esempio: *Bicycle*, *Car*, *Motorcycle*, *Truck*, o *Walking*

### Limitazioni

- Se si specifica `Walking` la modalità di viaggio e il provider di dati è Esri, la partenza e la destinazione devono essere entro 40 km.
- `Bicycle` o `Motorcycle` sono disponibili solo quando si utilizza Grab come fornitore di dati.
- Grab fornisce `Walking` percorsi solo `Bicycle` in determinate città. Per ulteriori informazioni, consulta [Paesi/regioni e aree coperte](#).
- `Truck` non è disponibile quando si utilizza Grab come fornitore di dati.

### Preferenze aggiuntive

Se si specifica un `TravelMode` di *Car*, è possibile specificare preferenze di percorso aggiuntive con il seguente parametro opzionale:

- `CarModeOptions`— specifica le preferenze relative al percorso quando si viaggia in auto, ad esempio *AvoidFerries* e *AvoidTolls*.

Se si specifica un `TravelMode` di *Truck*, è possibile specificare preferenze di percorso aggiuntive con il seguente parametro opzionale:

- `TruckModeOptions`— Consente di specificare le preferenze relative al percorso quando si viaggia su un camion, ad esempio *AvoidFerries* e *AvoidTolls*, oltre a specificare i percorsi che possono accogliere il *TruckDimensionstreno*. *TruckWeight*

## Example

La seguente [CalculateRoute](#) richiesta specifica *Truck* come modalità di viaggio. Le restrizioni aggiuntive sugli itinerari includono: evitare percorsi che utilizzano traghetti ed evitare strade che non possono ospitare le dimensioni e il peso del camion.

```
{
  "DeparturePosition": [-122.7565,49.0021],
  "DestinationPosition": [-122.3394, 47.6159],
  "DepartNow": true,
  "TravelMode": "Truck",
  "TruckModeOptions": {
    "AvoidFerries": true,
    "AvoidTolls": false,
    "Dimensions": {
      "Height": 4.5,
      "Length": 15.5,
      "Unit": "Meters",
      "Width": 4.5
    },
    "Weight": {
      "Total": 7500,
      "Unit": "Pounds"
    }
  }
}
```

## Gestione delle risorse del calcolatore di percorso

Puoi gestire le risorse del calcolatore di percorso utilizzando la console Amazon LocationAWS CLI, o le API di Amazon Location.

### Elenca le risorse del tuo calcolatore di percorso

Puoi visualizzare l'elenco del calcolatore di percorso utilizzando la console Amazon LocationAWS CLI, o le API di Amazon Location:

#### Console

Per visualizzare un elenco di calcolatori di percorsi utilizzando la console Amazon Location

1. Apri la console Amazon Location all'[indirizzo https://console.aws.amazon.com/location/](https://console.aws.amazon.com/location/).
2. Scegli Calcolatori di percorso dal riquadro di navigazione a sinistra.
3. Visualizza i dettagli del calcolatore del percorso in I miei calcolatori di percorso.

#### API

Usa l'[ListRouteCalculators](#) operazione dalle API Amazon Location Routes.

L'esempio seguente è una richiesta API per ottenere un elenco di calcolatori di percorsi nell'AWSaccount.

```
POST /routes/v0/list-calculators
```

Di seguito è riportato un esempio di risposta per [ListRouteCalculators](#):

```
{
  "Entries": [
    {
      "CalculatorName": "ExampleCalculator",
      "CreateTime": 2020-09-30T22:59:34.142Z,
      "DataSource": "Esri",
      "Description": "string",
      "UpdateTime": 2020-09-30T23:59:34.142Z
    }
  ],
  "NextToken": "1234-5678-9012"
```

```
}
```

## CLI

Utilizza il comando `list-route-calculators`.

L'esempio seguente serve AWS CLI per ottenere un elenco di calcolatori di percorso presenti nell'AWSaccount.

```
aws location list-route-calculators
```

## Ottieni i dettagli del calcolatore del percorso

Puoi ottenere dettagli su qualsiasi risorsa di calcolo del percorso nel tuo AWS account utilizzando la console Amazon LocationAWS CLI, o le API di Amazon Location:

### Console

Per visualizzare i dettagli di un calcolatore di percorso utilizzando la console Amazon Location

1. Apri la console Amazon Location all'[indirizzo https://console.aws.amazon.com/location/](https://console.aws.amazon.com/location/).
2. Scegli Calcolatori di percorso dal riquadro di navigazione a sinistra.
3. In I miei calcolatori di percorso, seleziona il link del nome del calcolatore del percorso di destinazione.

### API

Usa l'[DescribeRouteCalculator](#)operazione dalle API Amazon Location Routes.

L'esempio seguente è una richiesta API per cui ottenere i dettagli del calcolatore di percorso.

*ExampleCalculator*

```
GET /routes/v0/calculators/ExampleCalculator
```

Di seguito è riportato un esempio di risposta per[DescribeRouteCalculator](#):

```
{
  "CalculatorArn": "arn:aws:geo:us-west-2:123456789012:route-
calculator/ExampleCalculator",
```



```
"CalculatorName": "ExampleCalculator",
"CreateTime": 2020-09-30T22:59:34.142Z,
"DataSource": "Esri",
"Description": "string",
"Tags": {
  "Tag1" : "Value1"
},
"UpdateTime": 2020-09-30T23:59:34.142Z
}
```

## CLI

Utilizza il comando `describe-route-calculator`.

L'esempio seguente è un AWS CLI modo per ottenere i dettagli del calcolatore di percorso per *ExampleCalculator*.

```
aws location describe-route-calculator \
  --calculator-name "ExampleCalculator"
```

## Eliminare un calcolatore di percorso

Puoi eliminare un calcolatore di percorso dal tuo AWS account utilizzando la console Amazon LocationAWS CLI, o le API di Amazon Location:

### Console

Per eliminare un calcolatore di percorso utilizzando la console Amazon Location

#### Warning

Questa operazione elimina la risorsa in modo permanente.

1. Apri la console Amazon Location all'[indirizzo https://console.aws.amazon.com/location/](https://console.aws.amazon.com/location/).
2. Scegli Calcolatori di percorso dal riquadro di navigazione a sinistra.
3. In I miei calcolatori di percorso, seleziona il calcolatore del percorso di destinazione.
4. Scegli Elimina il calcolatore del percorso.

## API

Usa l'[DeleteRouteCalculator](#) operazione dalle API Amazon Location Routes.

L'esempio seguente è una richiesta API per eliminare la raccolta geofence.

*ExampleCalculator*

```
DELETE /routes/v0/calculators/ExampleCalculator
```

Di seguito è riportato un esempio di risposta per: [DeleteRouteCalculator](#)

```
HTTP/1.1 200
```

## CLI

Utilizza il comando `delete-route-calculator`.

L'esempio seguente è un AWS CLI comando per eliminare la raccolta geofence.

*ExampleCalculator*

```
aws location delete-route-calculator \  
    --calculator-name "ExampleCalculator"
```

## Geofencing di un'area di interesse utilizzando Amazon Location

Un'applicazione di geofencing valuta la posizione di un dispositivo tracciato rispetto alle aree di interesse registrate in precedenza. Ciò consente di intraprendere azioni in base agli aggiornamenti della posizione. Ad esempio, puoi avviare un evento che richieda una notifica quando un cliente che ha ordinato un caffè sulla sua app mobile si trova vicino a un negozio.

### Note

Per una panoramica dei concetti di geofencing e tracker, vedi. [Geofence e tracker](#)

Questa sezione della guida fornisce step-by-step istruzioni per creare un'applicazione di geofencing utilizzando Amazon Location Service.

### Panoramica delle fasi

1. Aggiungi geofence attorno alle aree di interesse e archiviali in una risorsa di raccolta di geofence.
2. Inizia a tracciare i dispositivi bersaglio e archivia la cronologia delle posizioni del dispositivo in una risorsa di localizzazione.
3. Collega la tua risorsa di localizzazione alla tua risorsa di raccolta di geofence in modo che gli aggiornamenti sulla posizione del dispositivo vengano valutati automaticamente rispetto a tutti i geofence.
4. Puoi valutare le posizioni dei dispositivi direttamente rispetto alle risorse di raccolta dei geofence se non desideri utilizzare Amazon Location Trackers per conservare la cronologia delle posizioni dei tuoi dispositivi.

Dopo aver implementato la soluzione di geofencing, la risorsa di raccolta di geofence emette i seguenti eventi:

- ENTER— Un dispositivo tracciato entra in un geofence all'interno di una raccolta di geofence.
- EXIT— Un dispositivo tracciato esce da un geofence all'interno di una raccolta di geofence.

Puoi usare Amazon EventBridge per reagire agli eventi indirizzandoli altrove.

In alternativa all'invio di aggiornamenti tramite le API di Amazon Location Service da ciascun dispositivo, puoi utilizzare MQTT per inviare gli aggiornamenti dei dispositivi.

I seguenti argomenti descrivono in dettaglio questi passaggi e le alternative.

## Argomenti

- [Aggiungi geofence](#)
- [Inizia il tracciamento](#)
- [Collega un tracker a una raccolta di geofence](#)
- [Valuta le posizioni dei dispositivi rispetto ai geofence](#)
- [Verifica le posizioni dei dispositivi](#)
- [Reagire agli eventi di Amazon Location Service con Amazon EventBridge](#)
- [Monitoraggio tramite MQTT AWS IoT e Amazon Location Service](#)
- [Gestione delle risorse di raccolta di geofence](#)
- [Gestione delle risorse del tracker](#)
- [Esempio di applicazione mobile Geofencing and Tracking](#)

## Aggiungi geofence

I geofence contengono punti e vertici che formano un confine chiuso, che definisce un'area di interesse. Le collezioni Geofence archiviano e gestiscono uno o più geofence.

[Le raccolte di geofence di Amazon Location archiviano le geofence definite utilizzando un formato di dati geospaziali standard chiamato GeoJSON \(RFC 7946\)](#). Puoi utilizzare strumenti, come [geojson.io](#), gratuitamente per disegnare graficamente i tuoi geofence e salvare il file GeoJSON di output.

### Note

Amazon Location non supporta poligoni con fori, multipoligoni, poligoni in senso orario e geofence che attraversano l'antimeridiano.

## Crea una raccolta di geofence

Crea una raccolta di geofence per archiviare e gestire i geofence utilizzando la console Amazon Location, AWS CLI o le API di Amazon Location.

### Console

Per creare una raccolta di geofence utilizzando la console Amazon Location

1. Apri la console Amazon Location Service all'[indirizzo https://console.aws.amazon.com/location/](https://console.aws.amazon.com/location/).
2. Nel riquadro di navigazione a sinistra, scegli Collezioni Geofence.
3. Scegli Crea collezione geofence.
4. Compila le seguenti caselle:
  - Nome: inserisci un nome univoco. Ad esempio, *ExampleGeofenceCollection*. Massimo 100 caratteri. Le voci valide includono caratteri alfanumerici, trattini, punti e caratteri di sottolineatura.
  - Descrizione: inserisci una descrizione opzionale per differenziare le tue risorse.
5. In base a EventBridge rule with CloudWatch as a target, puoi creare una EventBridge regola opzionale per iniziare a [reagire agli eventi di geofence](#). Ciò consente ad Amazon Location di pubblicare eventi su Amazon CloudWatch Logs.

6. (Facoltativo) In Tag, immettere Key (Chiave) e Value (Valore) di un tag. Questo aggiunge un tag alla tua nuova collezione di geofence. Per ulteriori informazioni, consulta [Taggare le risorse di Amazon Location Service](#).
7. (Facoltativo) In Crittografia a chiave gestita dal cliente, puoi scegliere di aggiungere una chiave gestita dal cliente. Questo aggiunge una chiave simmetrica gestita dal cliente che puoi creare, possedere e gestire tramite la crittografia predefinita AWS di proprietà. Per ulteriori informazioni, consulta [Crittografia dei dati](#) inattivi.
8. Scegli Crea raccolta di geofence.

## API

Per creare una raccolta di geofence utilizzando le API di Amazon Location

Usa l'[CreateGeofenceCollection](#) operazione dalle API Amazon Location Geofences.

L'esempio seguente utilizza una richiesta API per creare una raccolta di geofence chiamata *ExampleGeofenceCollection*. La raccolta di geofence è associata a una [AWS KMS chiave gestita dal cliente per crittografare](#) i dati dei clienti.

```
POST /geofencing/v0/collections
Content-type: application/json

{
  "CollectionName": "ExampleGeofenceCollection",
  "Description": "Geofence collection 1 for shopping center",
  "KmsKeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
  "Tags": {
    "Tag1" : "Value1"
  }
}
```

## AWS CLI

Per creare una raccolta di geofence utilizzando i comandi AWS CLI

Utilizza il comando [create-geofence-collection](#).

L'esempio seguente utilizza un AWS CLI per creare una raccolta di geofence chiamata *ExampleGeofenceCollection*. La raccolta di geofence è associata a una [AWS KMS chiave gestita dal cliente per crittografare](#) i dati dei clienti.

```
aws location \  
  create-geofence-collection \  
  --collection-name "ExampleGeofenceCollection" \  
  --description "Shopping center geofence collection" \  
  --kms-key-id "1234abcd-12ab-34cd-56ef-1234567890ab" \  
  --tags Tag1=Value1
```

### Note

La fatturazione dipende dall'utilizzo. Potresti incorrere in commissioni per l'uso di altri servizi. AWS Per ulteriori informazioni, consulta i [prezzi di Amazon Location Service](#).

## Disegna geofence

Ora che hai creato la tua collezione di geofence, puoi definire i tuoi geofence. I geofence sono definiti come poligono o come cerchio. [Per disegnare un geofence poligonale puoi usare uno strumento di modifica GeoJSON, come geojson.io.](#)

Per creare un geofence come cerchio, è necessario definire il punto centrale del cerchio e il raggio. Ad esempio, se si desidera creare un geofence per ricevere una notifica ogni volta che un dispositivo si trova a 50 metri da una posizione specifica, è necessario utilizzare la latitudine e la longitudine di tale posizione e specificare il raggio di 50 metri.

Utilizzando le API di Amazon Location Service, puoi anche aggiungere metadati al tuo geofence, sotto forma di coppie chiave-valore. Questi possono essere utili per archiviare informazioni sul geofence, ad esempio il tipo, o altre informazioni specifiche dell'applicazione. Puoi utilizzare questi metadati quando. [Reagire agli eventi di Amazon Location Service con Amazon EventBridge](#)

## Aggiungere geofence poligonali

Questa sezione descrive la creazione di geofence poligonali

Disegna geofence usando uno strumento GeoJSON

[Ora che hai creato la tua collezione di geofence, puoi definire i tuoi geofence utilizzando uno strumento di modifica GeoJSON, come geojson.io.](#)

Per creare un file GeoJSON

1. Apri uno strumento di modifica GeoJSON. Ad esempio, [geojson.io](https://geojson.io).
2. Scegli l'icona Disegna un poligono e disegna la tua area di interesse.
3. Scegli Salva, quindi scegli GeoJSON dal menu a discesa.

Inserisci geofence GeoJSON in una raccolta di geofence

Puoi utilizzare il file GeoJSON risultante per caricare i tuoi geofence utilizzando la console Amazon Location Service, o AWS CLI le API di localizzazione di Amazon:

## Console

Per aggiungere un geofence a una raccolta di geofence utilizzando la console Amazon Location Service

1. Apri la console Amazon Location Service all'[indirizzo https://console.aws.amazon.com/location/](https://console.aws.amazon.com/location/).
2. Nel riquadro di navigazione a sinistra, scegli Collezioni Geofence.
3. Dall'elenco delle raccolte Geofence, seleziona il link del nome per la raccolta di geofence di destinazione.
4. In Geofences, scegli Crea geofences.
5. Nella finestra Aggiungi geofences, trascina e rilascia il tuo GeoJSON nella finestra.
6. Scegli Aggiungi geofence.

## API

Per aggiungere geofence utilizzando le API di Amazon Location

Usa l'[PutGeofence](#) operazione dalle API Amazon Location Geofences.

L'esempio seguente utilizza una richiesta API per aggiungere un geofence con l'ID GEOFENCE-EXAMPLE1 a una raccolta *di geofence* chiamata *ExampleGeofenceCollection*. Inoltre, specifica una singola proprietà dei metadati del geofence con la chiave e il valore. Type `loadingArea`

```
PUT /geofencing/v0/collections/ExampleGeofenceCollection/geofence/GEOFENCE-EXAMPLE1
Content-type: application/json
```

```

{
  "GeofenceProperties": {
    "Type" : "loadingArea"
  },
  "Geometry": {
    "Polygon": [
      [
        [-5.716667, -15.933333],
        [-14.416667, -7.933333],
        [-12.316667, -37.066667],
        [-5.716667, -15.933333]
      ]
    ]
  }
}

```

In alternativa, è possibile aggiungere più di un geofence utilizzando l'operazione.

### [BatchPutGeofence](#)

```

POST /geofencing/v0/collections/ExampleGeofenceCollection/put-geofences
Content-type: application/json

```

```

{
  "Entries": [
    {
      "GeofenceProperties": {
        "Type" : "loadingArea"
      },
      "GeofenceId": "GEOFENCE-EXAMPLE1",
      "Geometry": {
        "Polygon": [
          [
            [-5.716667, -15.933333],
            [-14.416667, -7.933333],
            [-12.316667, -37.066667],
            [-5.716667, -15.933333]
          ]
        ]
      }
    }
  ]
}

```



## AWS CLI

Per aggiungere un geofence a una raccolta di geofence utilizzando i comandi AWS CLI

Utilizza il comando [put-geofence](#).

L'esempio seguente utilizza un AWS CLI per aggiungere un geofence a una raccolta di geofence chiamata. *ExampleGeofenceCollection*

```
$ aws location \
  put-geofence \
    --collection-name ExampleGeofenceCollection \
    --geofence-id ExampleGeofenceTriangle \
    --geofence-properties '{"Type": "loadingArea"}' \
    --geometry 'Polygon=[[[[-5.716667, -15.933333],[-14.416667, -7.933333],
[-12.316667, -37.066667],[-5.716667, -15.933333]]]]'
{
  "CreateTime": "2020-11-11T00:16:14.487000+00:00",
  "GeofenceId": "ExampleGeofenceTriangle",
  "UpdateTime": "2020-11-11T00:19:59.894000+00:00"
}
```

## Aggiungere geofence circolari

Questa sezione descrive la creazione di geofence circolari. È necessario conoscere la latitudine e la longitudine del punto in cui si desidera che sia il centro del cerchio e il raggio in metri del cerchio. Puoi creare geofence circolari con le API di Amazon Location o il AWS CLI

### API

Per aggiungere geofence circolari utilizzando le API di Amazon Location

Usa l'[PutGeofence](#) operazione dalle API Amazon Location Geofences.

L'esempio seguente utilizza una richiesta API per aggiungere un geofence con l'ID GEOFENCE-EXAMPLE2 a una raccolta *di geofence* chiamata: *ExampleGeofenceCollection*

```
PUT /geofencing/v0/collections/ExampleGeofenceCollection/geofence/GEOFENCE-EXAMPLE2
Content-type: application/json
```

```
{
  "Geometry": {
    "Circle": {
      "Center": [-5.716667, -15.933333],
      "Radius": 50
    }
  }
}
```

## AWS CLI

Per aggiungere un geofence circolare a una raccolta di geofence utilizzando i comandi AWS CLI

Utilizza il comando [put-geofence](#).

L'esempio seguente utilizza un AWS CLI per aggiungere un geofence a una raccolta di geofence chiamata *ExampleGeofenceCollection*

```
$ aws location \
  put-geofence \
    --collection-name ExampleGeofenceCollection \
    --geofence-id ExampleGeofenceCircle \
    --geometry 'Circle={Center=[-5.716667, -15.933333], Radius=50}'
```

### Note

È inoltre possibile inserire JSON per la geometria complessa in un file separato, come nell'esempio seguente.

```
$ aws location \
  put-geofence \
    --collection-name ExampleGeofenceCollection \
    --geofence-id ExampleGeofenceCircle \
    --geometry file:circle.json
```

Nell'esempio, il file `circle.json` include JSON per la geometria del cerchio.

```
{
  "Circle": {
    "Center": [-74.006975, 40.717127],
    "Radius": 287.7897969218057
  }
}
```

```
}  
}
```

## Inizia il tracciamento

Questa sezione ti guida nella creazione di un'applicazione di tracciamento che acquisisca le posizioni dei dispositivi.

### Crea un tracker

Crea una risorsa di tracciamento per archiviare ed elaborare gli aggiornamenti di posizione dai tuoi dispositivi. Puoi utilizzare la console Amazon Location Service AWS CLI, o le API Amazon Location.

Ogni aggiornamento di posizione memorizzato nelle risorse del tracker può includere una misura della precisione della posizione e fino a tre campi di metadati sulla posizione o sul dispositivo che desideri memorizzare. I metadati vengono memorizzati come coppie chiave-valore e possono memorizzare informazioni come velocità, direzione, pressione dei pneumatici o temperatura del motore.

I tracker filtrano gli aggiornamenti di posizione man mano che vengono ricevuti. Ciò riduce il rumore visivo nei percorsi del dispositivo (chiamato jitter) e il numero di falsi eventi di ingresso e uscita da geofence. Ciò aiuta anche a gestire i costi riducendo il numero di valutazioni di geofence avviate.

I tracker offrono tre opzioni di filtraggio della posizione per aiutare a gestire i costi e ridurre il jitter negli aggiornamenti delle posizioni.

- **Basato sulla precisione:** utilizzabile con qualsiasi dispositivo che fornisce una misurazione di precisione. La maggior parte dei dispositivi mobili fornisce queste informazioni. La precisione di ogni misurazione della posizione è influenzata da molti fattori ambientali, tra cui la ricezione satellitare GPS, il paesaggio e la vicinanza dei dispositivi Wi-Fi e Bluetooth. La maggior parte dei dispositivi, inclusa la maggior parte dei dispositivi mobili, è in grado di fornire una stima dell'accuratezza della misurazione insieme alla misurazione. Con AccuracyBased il filtraggio, Amazon Location ignora gli aggiornamenti sulla posizione se il dispositivo si è spostato meno della precisione misurata. Ad esempio, se due aggiornamenti consecutivi da un dispositivo hanno un intervallo di precisione compreso tra 5 m e 10 m, Amazon Location ignora il secondo aggiornamento se il dispositivo si è spostato a meno di 15 m. Amazon Location non valuta gli aggiornamenti ignorati rispetto ai geofence, né li archivia.

Se la precisione non viene fornita, viene considerata pari a zero e la misurazione è considerata perfettamente accurata.

#### Note

È inoltre possibile utilizzare filtri basati sulla precisione per rimuovere tutti i filtri. Se selezioni il filtraggio basato sulla precisione, ma sostituisci tutti i dati di precisione a zero o ometti completamente la precisione, Amazon Location non filtrerà alcun aggiornamento.

- **Basato sulla distanza:** utilizzalo quando i tuoi dispositivi non forniscono una misurazione di precisione, ma desideri comunque sfruttare il filtraggio per ridurre il jitter e gestire i costi. `DistanceBased` il filtraggio ignora gli aggiornamenti sulla posizione in cui i dispositivi si sono spostati a meno di 30 m (98,4 piedi). Quando utilizzi il filtro di `DistanceBased` posizione, Amazon Location non valuta questi aggiornamenti ignorati rispetto ai geofence né archivia gli aggiornamenti.

La precisione della maggior parte dei dispositivi mobili, inclusa la precisione media dei dispositivi iOS e Android, è entro 15 m. Nella maggior parte delle applicazioni, il `DistanceBased` filtraggio può ridurre l'effetto delle imprecisioni di posizione quando si visualizza la traiettoria del dispositivo su una mappa e l'effetto di rimbalzo di più eventi consecutivi di entrata e uscita quando i dispositivi si trovano vicino al confine di un geofence. Può anche contribuire a ridurre il costo dell'applicazione, effettuando meno chiamate per effettuare valutazioni sulla base di geofence collegati o recuperare le posizioni dei dispositivi.

- **Basato sul tempo:** (impostazione predefinita) Utilizzalo quando i dispositivi inviano aggiornamenti di posizione molto frequentemente (più di una volta ogni 30 secondi) e desideri ottenere valutazioni dei geofence quasi in tempo reale senza archiviare ogni aggiornamento. Nel `TimeBased` filtraggio, ogni aggiornamento di posizione viene valutato sulla base di raccolte di geofence collegate, ma non tutti gli aggiornamenti di posizione vengono archiviati. Se la frequenza di aggiornamento è superiore a 30 secondi, viene archiviato un solo aggiornamento ogni 30 secondi per ogni ID dispositivo univoco.

#### Note

Fai attenzione ai costi della tua applicazione di tracciamento quando decidi il metodo di filtraggio e la frequenza degli aggiornamenti delle posizioni. Ti verrà addebitato ogni aggiornamento della posizione e una volta per la valutazione dell'aggiornamento della

posizione rispetto a ciascuna raccolta di geofence collegata. Ad esempio, quando si utilizza il filtro basato sul tempo, se il tracker è collegato a due raccolte di geofence, ogni aggiornamento di posizione verrà conteggiato come una richiesta di aggiornamento della posizione e due valutazioni della raccolta di geofence. Se segnali aggiornamenti di posizione ogni 5 secondi per i tuoi dispositivi e utilizzi il filtro basato sul tempo, ti verranno fatturati 720 aggiornamenti della posizione e 1.440 valutazioni di geofence all'ora per ciascun dispositivo. La tua fattura non è influenzata dal numero di geofence presenti in ciascuna raccolta. Poiché ogni raccolta di geofence può contenere fino a 50.000 geofence, potresti voler combinare i geofence in un numero inferiore di raccolte, ove possibile, per ridurre i costi delle valutazioni dei geofence.

Per impostazione predefinita, riceverai EventBridge eventi ogni volta che un dispositivo tracciato entra o esce da un geofence collegato. Per ulteriori informazioni, consulta [Collega un tracker a una raccolta di geofence](#).

Puoi abilitare gli eventi per tutti gli aggiornamenti di posizione filtrati per una risorsa tracker. Per ulteriori informazioni, consulta [Abilita gli eventi di aggiornamento per un tracker](#).

#### Note

Se desideri crittografare i tuoi dati utilizzando la tua chiave gestita AWS KMS dal cliente, la funzione Bounding Polygon Queries sarà disabilitata per impostazione predefinita. Questo perché utilizzando questa funzione Bounding Polygon Queries, una rappresentazione delle posizioni del dispositivo non verrà crittografata utilizzando la chiave gestita. AWS KMS Tuttavia, la posizione esatta del dispositivo viene ancora crittografata utilizzando la chiave gestita.


Puoi scegliere di attivare la funzione Bounding Polygon Queries impostando il `KmsKeyEnableGeospatialQueries` parametro su `true` durante la creazione o l'aggiornamento di un Tracker.

## Console

Per creare un tracker utilizzando la console Amazon Location

1. Apri la console Amazon Location Service all'[indirizzo https://console.aws.amazon.com/location/](https://console.aws.amazon.com/location/).

2. Nel riquadro di navigazione a sinistra, scegli Trackers.
3. Scegli Crea tracker.
4. Compila i seguenti campi:
  - Nome: inserisci un nome univoco. Ad esempio, *ExampleTracker*. Massimo 100 caratteri. Le voci valide includono caratteri alfanumerici, trattini, punti e caratteri di sottolineatura.
  - Descrizione: immettere una descrizione facoltativa.
5. In Filtro di posizione, scegli l'opzione più adatta al modo in cui intendi utilizzare la risorsa del tracker. Se non imposti il filtro di posizione, l'impostazione predefinita è TimeBased Per ulteriori informazioni, [Localizzatori](#) consulta questa guida e [PositionFiltering](#) l'Amazon Location Service Trackers API Reference.
6. (Facoltativo) In Tag, immettere Key (Chiave) e Value (Valore) di un tag. Questo aggiunge un tag alla tua nuova collezione di geofence. Per ulteriori informazioni, consulta [Tagging delle risorse](#).
7. (Facoltativo) In Crittografia a chiave gestita dal cliente, puoi scegliere di aggiungere una chiave gestita dal cliente. Questo aggiunge una chiave simmetrica gestita dal cliente che puoi creare, possedere e gestire tramite la crittografia predefinita AWS di proprietà. Per ulteriori informazioni, consulta [Crittografia dei dati](#) inattivi.
8. (Facoltativo) In KmsKeyEnableGeospatialQueries, puoi scegliere di abilitare le interrogazioni geospaziali. Ciò consente di utilizzare la funzionalità Bounding Polygon Queries, crittografando i dati utilizzando una chiave gestita AWS KMS del cliente.

 Note

Quando utilizzi la funzione Bounding Polygon Queries, una rappresentazione delle posizioni del dispositivo non viene crittografata utilizzando la chiave gestita. AWS KMS Tuttavia, la posizione esatta del dispositivo viene ancora crittografata utilizzando la chiave gestita.

9. (Facoltativo) In EventBridge configurazione, puoi scegliere di abilitare EventBridge gli eventi per gli aggiornamenti delle posizioni filtrati. Questo invierà un evento ogni volta che un aggiornamento di posizione per un dispositivo in questo tracker soddisfa la valutazione del filtro di posizione.
10. Scegli Crea tracker.

## API

Per creare un tracker utilizzando le API di localizzazione di Amazon

Usa l'[CreateTracker](#) operazione dalle API di Amazon Location Trackers.

L'esempio seguente utilizza una richiesta API per creare un tracker chiamato. *ExampleTracker*  
La risorsa tracker è associata a una [AWS KMS chiave gestita dal cliente per crittografare i dati dei clienti](#) e non [consente l'aggiornamento delle posizioni. EventBridge](#)

```
POST /tracking/v0/trackers
Content-type: application/json

{
  "TrackerName": "ExampleTracker",
  "Description": "string",
  "KmsKeyEnableGeospatialQueries": false,
  "EventBridgeEnabled": false,
  "KmsKeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
  "PositionFiltering": "AccuracyBased",
  "Tags": {
    "string" : "string"
  }
}
```

### Crea un tracker con abilitato **KmsKeyEnableGeospatialQueries**

L'esempio seguente ha il parametro `KmsKeyEnableGeospatialQueries` impostato su `true`. Ciò consente di utilizzare la funzionalità Bounding Polygon Queries, crittografando i dati utilizzando una chiave gestita dal cliente. AWS KMS

Per informazioni sull'utilizzo della funzione Bounding Polygon Queries, vedi [???](#)

#### Note

Quando utilizzate la funzione Bounding Polygon Queries, una rappresentazione delle posizioni del dispositivo non viene crittografata utilizzando la chiave gestita. AWS KMS  
Tuttavia, la posizione esatta del dispositivo viene ancora crittografata utilizzando la chiave gestita.

```
POST /tracking/v0/trackers
Content-type: application/json

{
  "TrackerName": "ExampleTracker",
  "Description": "string",
  "KmsKeyEnableGeospatialQueries": true,
  "EventBridgeEnabled": false,
  "KmsKeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
  "PositionFiltering": "AccuracyBased",
  "Tags": {
    "string" : "string"
  }
}
```

## AWS CLI

Per creare un tracker utilizzando AWS CLI i comandi

Utilizza il comando [create-tracker](#).

L'esempio seguente utilizza il AWS CLI per creare un tracker chiamato. *ExampleTracker* La risorsa tracker è associata a una [AWS KMS chiave gestita dal cliente per crittografare i dati dei clienti](#) e non [consente l'aggiornamento delle posizioni](#). [EventBridge](#)

```
aws location \
  create-tracker \
  --tracker-name "ExampleTracker" \
  --position-filtering "AccuracyBased" \
  --event-bridge-enabled false \
  --kms-key-enable-geospatial-queries false \
  --kms-key-id "1234abcd-12ab-34cd-56ef-1234567890ab"
```

### Crea un tracker con abilitato **KmsKeyEnableGeospatialQueries**

L'esempio seguente ha il parametro `KmsKeyEnableGeospatialQueries` impostato su `true`. Ciò consente di utilizzare la funzionalità Bounding Polygon Queries, crittografando i dati utilizzando una chiave gestita dal cliente. AWS KMS

Per informazioni sull'utilizzo della funzione Bounding Polygon Queries, vedi [???](#)



**Note**

Quando utilizzate la funzione Bounding Polygon Queries, una rappresentazione delle posizioni del dispositivo non viene crittografata utilizzando la chiave gestita. AWS KMS Tuttavia, la posizione esatta del dispositivo viene ancora crittografata utilizzando la chiave gestita.

```
aws location \  
  create-tracker \  
    --tracker-name "ExampleTracker" \  
    --position-filtering "AccuracyBased" \  
    --event-bridge-enabled false \  
    --kms-key-enable-geospatial-queries true \  
    --kms-key-id "1234abcd-12ab-34cd-56ef-1234567890ab"
```

**Note**

La fatturazione dipende dall'utilizzo. Potresti incorrere in commissioni per l'uso di altri servizi. AWS Per ulteriori informazioni, consulta i [prezzi di Amazon Location Service](#).

Puoi modificare la descrizione, il filtro della posizione e la EventBridge configurazione dopo aver creato il tracker scegliendo Modifica tracker.

## Autenticazione delle tue richieste

Una volta creata una risorsa di tracciamento e pronto per iniziare a valutare le posizioni dei dispositivi rispetto ai geofence, scegli come autenticare le tue richieste:

- Per scoprire come accedere ai servizi, consulta [Accedere ad Amazon Location Service](#).
- Se desideri pubblicare le posizioni dei dispositivi con richieste non autenticate, puoi utilizzare Amazon Cognito.

### Esempio

L'esempio seguente mostra l'utilizzo di un pool di identità Amazon Cognito per l'autorizzazione, l'utilizzo di [AWS JavaScript SDK v3](#) e Amazon Location. [JavaScript Aiutante di autenticazione](#)

```
import { LocationClient, BatchUpdateDevicePositionCommand } from "@aws-sdk/client-
location";
import { withIdentityPoolId } from "@aws/amazon-location-utilities-auth-helper";

// Unauthenticated identity pool you created
const identityPoolId = "us-east-1:1234abcd-5678-9012-abcd-sample-id";

// Create an authentication helper instance using credentials from Cognito
const authHelper = await withIdentityPoolId(identityPoolId);

const client = new LocationClient({
  region: "us-east-1", // The region containing both the identity pool and tracker
  resource
  ...authHelper.getLocationClientConfig(), // Provides configuration required to make
  requests to Amazon Location
});

const input = {
  TrackerName: "ExampleTracker",
  Updates: [
    {
      DeviceId: "ExampleDevice-1",
      Position: [-123.4567, 45.6789],
      SampleTime: new Date("2020-10-02T19:09:07.327Z"),
    },
    {
      DeviceId: "ExampleDevice-2",
      Position: [-123.123, 45.123],
      SampleTime: new Date("2020-10-02T19:10:32Z"),
    },
  ],
};

const command = new BatchUpdateDevicePositionCommand(input);

// Send device position updates
const response = await client.send(command);
```

## Aggiorna il tracker con la posizione del dispositivo

Per tracciare i tuoi dispositivi, puoi pubblicare aggiornamenti sulla posizione del dispositivo sul tracker. In un secondo momento puoi recuperare queste posizioni del dispositivo o la cronologia delle posizioni del dispositivo dalla tua risorsa tracker.

Ogni aggiornamento della posizione deve includere l'ID del dispositivo, un timestamp e una posizione. Facoltativamente, puoi includere altri metadati, tra cui la precisione e fino a 3 coppie chiave-valore per uso personale.

Se il tracker è collegato a una o più raccolte di geofence, gli aggiornamenti verranno valutati rispetto a tali geofence (seguendo le regole di filtro che hai specificato per il tracker). Se un dispositivo viola un'area geofrecintata (spostandosi dall'interno all'esterno dell'area o viceversa), riceverai eventi al suo interno. EventBridge Questi ENTER o questi EXIT eventi includono i dettagli dell'aggiornamento della posizione, tra cui l'ID del dispositivo, il timestamp e tutti i metadati associati.

### Note

Per ulteriori informazioni sul filtraggio delle posizioni, vedere. [Crea un tracker](#)  
Per ulteriori informazioni sugli eventi di geofence, vedere. [Reagire agli eventi di Amazon Location Service con Amazon EventBridge](#)

Utilizza uno di questi metodi per inviare gli aggiornamenti del dispositivo:

- [Invia aggiornamenti MQTT](#) a una risorsa AWS IoT Core e collegala alla tua risorsa tracker.
- Invia aggiornamenti sulla posizione utilizzando l'API Amazon Location Trackers, utilizzando o le AWS CLI API Amazon Location. Puoi utilizzare gli [AWS SDK](#) per chiamare le API dalla tua applicazione iOS o Android.

## API

Per inviare un aggiornamento di posizione utilizzando le API di Amazon Location

Usa l'[BatchUpdateDevicePosition](#) operazione dalle API di Amazon Location Trackers.

L'esempio seguente utilizza una richiesta API per pubblicare un aggiornamento della posizione del dispositivo su un tracker *ExampleDevice*. *ExampleTracker*

```
POST /tracking/v0/trackers/ExampleTracker/positions
```

```
Content-type: application/json
{
  "Updates": [
    {
      "DeviceId": "1",
      "Position": [
        -123.12245146162303, 49.27521118043802
      ],
      "SampleTime": "2022-10-24T19:09:07.327Z",
      "PositionProperties": {
        "name" : "device1"
      },
      "Accuracy": {
        "Horizontal": 10
      }
    },
    {
      "DeviceId": "2",
      "Position": [
        -123.1230104928471, 49.27752402723152
      ],
      "SampleTime": "2022-10-02T19:09:07.327Z"
    },
    {
      "DeviceId": "3",
      "Position": [
        -123.12325592118916, 49.27340530543111
      ],
      "SampleTime": "2022-10-02T19:09:07.327Z"
    },
    {
      "DeviceId": "4",
      "Position": [
        -123.11958813096311, 49.27774641063121
      ],
      "SampleTime": "2022-10-02T19:09:07.327Z"
    },
    {
      "DeviceId": "5",
      "Position": [
        -123.1277418058896, 49.2765989015285
      ],
      "SampleTime": "2022-10-02T19:09:07.327Z"
    }
  ]
}
```

```
    },  
    {  
      "DeviceId": "6",  
      "Position": [  
        -123.11964267059481, 49.274188155916534  
      ],  
      "SampleTime": "2022-10-02T19:09:07.327Z"  
    }  
  ]  
}
```

## AWS CLI

Per inviare un aggiornamento della posizione utilizzando i comandi AWS CLI

Utilizza il comando [batch-update-device-position](#).

L'esempio seguente utilizza un AWS CLI per inviare a un tracker *ExampleTracker* un aggiornamento della posizione del dispositivo per *ExampleDevice-1* e *ExampleDevice-2*.

```
aws location batch-update-device-position \  
--tracker-name ExampleTracker \  
--updates '[{"DeviceId":"ExampleDevice-1","Position":  
[-123.123,47.123],"SampleTime":"2021-11-30T21:47:25.149Z"},  
{"DeviceId":"ExampleDevice-2","Position":  
[-123.123,47.123],"SampleTime":"2021-11-30T21:47:25.149Z","Accuracy":  
{"Horizontal":10.30},"PositionProperties":{"field1":"value1","field2":"value2"}}]'
```

## Ottieni la cronologia delle posizioni di un dispositivo da un tracker

La tua risorsa Amazon Location Tracker conserva la cronologia delle posizioni di tutti i tuoi dispositivi tracciati per un periodo di 30 giorni. Puoi recuperare la cronologia delle posizioni del dispositivo, inclusi tutti i metadati associati, dalla tua risorsa di localizzazione. I seguenti esempi utilizzano le AWS CLI, o le API Amazon Location.

### API

Per ottenere la cronologia delle posizioni del dispositivo da un tracker utilizzando le API di localizzazione di Amazon

Usa l'[GetDevicePositionHistory](#) operazione dalle API di Amazon Location Trackers.

L'esempio seguente utilizza una richiesta URI API per ottenere la cronologia delle posizioni del dispositivo *ExampleDevice* da un tracker chiamato *ExampleTracker* starting from 19:05:07 (inclusive) e termina con 19:20:07 (exclusive) on. 2020-10-02

```
POST /tracking/v0/trackers/ExampleTracker/devices/ExampleDevice/list-positions
Content-type: application/json
{
  "StartTimeInclusive": "2020-10-02T19:05:07.327Z",
  "EndTimeExclusive": "2020-10-02T19:20:07.327Z"
}
```

## AWS CLI

Per ottenere la cronologia delle posizioni del dispositivo da un tracker utilizzando i comandi AWS CLI

Utilizza il comando [get-device-position-history](#).

L'esempio seguente utilizza an AWS CLI per ottenere la cronologia delle posizioni del dispositivo *ExampleDevice* da un tracker chiamato *ExampleTracker* a partire da 19:05:07 (incluso) e termina con 19:20:07 (esclusivo) on. 2020-10-02

```
aws location \
  get-device-position-history \
    --device-id "ExampleDevice" \
    --start-time-inclusive "2020-10-02T19:05:07.327Z" \
    --end-time-exclusive "2020-10-02T19:20:07.327Z" \
    --tracker-name "ExampleTracker"
```

## Elenca le posizioni del dispositivo

Puoi visualizzare un elenco delle posizioni dei dispositivi per un tracker utilizzando o le API Amazon Location con l' `ListDevicePositions` API. AWS CLI Quando chiami l' `ListDevicePositions` API, viene restituito un elenco delle posizioni più recenti per tutti i dispositivi associati a un determinato tracker. Per impostazione predefinita, questa API restituisce 100 delle ultime posizioni del dispositivo per pagina di risultati per un determinato tracker. Per restituire solo i dispositivi all'interno di una regione specifica, utilizza il `FilterGeometry` parametro per creare una `Bounding Polygon Query`. In questo modo, quando si chiama `ListDevicePositions`, verranno restituiti solo i dispositivi all'interno del poligono.

### Note

Se desideri crittografare i tuoi dati utilizzando la tua chiave gestita AWS KMS dal cliente, la funzione Bounding Polygon Queries sarà disabilitata per impostazione predefinita. Questo perché utilizzando questa funzione, una rappresentazione delle posizioni del dispositivo non verrà crittografata utilizzando la chiave gestita. AWS KMS La posizione esatta del dispositivo, tuttavia, è ancora crittografata utilizzando la chiave gestita.

Puoi scegliere di attivare la funzione Bounding Polygon Queries. Questo viene fatto impostando il `KmsKeyEnableGeospatialQueries` parametro su `true` durante la creazione o l'aggiornamento di un Tracker.

## API

Usa l'[ListDevicePositions](#) operazione dalle API di Amazon Location Trackers.

L'esempio seguente è una richiesta API per ottenere un elenco di posizioni dei dispositivi nell'area poligonale, utilizzando il parametro opzionale. [FilterGeometry](#) L'esempio restituisce 3 posizioni dei dispositivi presenti nell'area definita dall'`Polygon` array.

```
POST /tracking/v0/trackers/TrackerName/list-positions HTTP/1.1
Content-type: application/json
```

```
{
  "FilterGeometry": {
    "Polygon": [
      [
        [
          -123.12003339442259,
          49.27425121147397
        ],
        [
          -123.1176984148229,
          49.277063620879744
        ],
        [
          -123.12389509145294,
          49.277954183760926
        ],
        [
          -123.12755921328647,
```

```
        49.27554025235713
      ],
      [
        -123.12330236586217,
        49.27211836076236
      ],
      [
        -123.12003339442259,
        49.27425121147397
      ]
    ]
  ],
  "MaxResults": 3,
  "NextToken": "1234-5678-9012"
}
```

Di seguito è riportato un esempio di risposta per [ListDevicePositions](#):

```
{
  "Entries": [
    {
      "DeviceId": "1",
      "SampleTime": "2022-10-24T19:09:07.327Z",
      "Position": [
        -123.12245146162303,
        49.27521118043802
      ],
      "Accuracy": {
        "Horizontal": 10
      },
      "PositionProperties": {
        "name": "device1"
      }
    },
    {
      "DeviceId": "3",
      "SampleTime": "2022-10-02T19:09:07.327Z",
      "Position": [
        -123.12325592118916,
        49.27340530543111
      ]
    }
  ],
}
```



```
{
  "DeviceId": "2",
  "SampleTime": "2022-10-02T19:09:07.327Z",
  "Position": [
    -123.1230104928471,
    49.27752402723152
  ]
},
"NextToken": "1234-5678-9012"
}
```

## CLI

Utilizza il comando [list-trackers](#).

L'esempio seguente serve AWS CLI per ottenere un elenco di dispositivi in un'area poligonale.

```
aws location list-device-positions TODO: add arguments add props for filter geo
```

## Collega un tracker a una raccolta di geofence

Ora che hai una raccolta di geofence e un tracker, puoi collegarli tra loro in modo che gli aggiornamenti di posizione vengano valutati automaticamente rispetto a tutti i geofence. Se non desideri valutare tutti gli aggiornamenti di posizione o, in alternativa, se non stai archiviando alcune delle tue posizioni in una risorsa di localizzazione, puoi [valutare](#) le posizioni dei dispositivi rispetto ai geofence su richiesta.

Quando le posizioni dei dispositivi vengono valutate rispetto ai geofence, vengono generati degli eventi. È possibile impostare un'azione per questi eventi. Per ulteriori informazioni sulle azioni che puoi impostare per gli eventi di geofence, consulta [Reagire agli eventi di Amazon Location Service con Amazon EventBridge](#)

Un evento Amazon Location include gli attributi dell'aggiornamento della posizione del dispositivo che lo genera e alcuni attributi del geofence che viene inserito o uscito. Per ulteriori informazioni sui dati inclusi in un evento di geofence, consulta. [Esempi di EventBridge eventi Amazon per Amazon Location Service](#)

I seguenti esempi collegano una risorsa tracker a una raccolta di geofence utilizzando la console, o le AWS CLI API Amazon Location.

## Console

Per collegare una risorsa tracker a una raccolta di geofence utilizzando la console Amazon Location Service

1. Apri la console Amazon Location Service all'[indirizzo https://console.aws.amazon.com/location/](https://console.aws.amazon.com/location/).
2. Nel riquadro di navigazione a sinistra, scegli Trackers.
3. In Device tracker, seleziona il link del nome del tracker di destinazione.
4. In Collezioni Geofence collegate, scegli Collega collezione Geofence.
5. Nella finestra Linked Geofence Collection, seleziona una collezione di geofence dal menu a discesa.
6. Scegliere Link (Collegamento).

Dopo aver collegato la risorsa tracker, ad essa verrà assegnato uno stato Attivo.

## API

Per collegare una risorsa tracker a una raccolta di geofence utilizzando le API di Amazon Location

Usa l'[AssociateTrackerConsumer](#) operazione dalle API di Amazon Location Trackers.

L'esempio seguente utilizza una richiesta API associata *ExampleTracker* a una raccolta di geofence utilizzando il relativo [Amazon Resource Name \(ARN\)](#).

```
POST /tracking/v0/trackers/ExampleTracker/consumers
Content-type: application/json

{
  "ConsumerArn": "arn:aws:geo:us-west-2:123456789012:geofence-
collection/ExampleGeofenceCollection"
}
```

## AWS CLI

Per collegare una risorsa tracker a una raccolta di geofence utilizzando i comandi AWS CLI

Utilizza il comando [associate-tracker-consumer](#).

L'esempio seguente utilizza un AWS CLI per creare una raccolta di geofence chiamata *ExampleGeofenceCollection*

```
aws location \  
  associate-tracker-consumer \  
    --consumer-arn "arn:aws:geo:us-west-2:123456789012:geofence-  
collection/ExampleGeofenceCollection" \  
    --tracker-name "ExampleTracker"
```

## Valuta le posizioni dei dispositivi rispetto ai geofence

Esistono due modi per valutare le posizioni rispetto ai geofence per generare eventi di geofence:

- Puoi collegare Trackers e Geofence Collections. Per maggiori informazioni, consulta la sezione: [Collega un tracker a una raccolta di geofence](#)
- Puoi fare una richiesta diretta alla risorsa di raccolta geofence per valutare una o più posizioni, utilizzando l'[BatchEvaluateGeofencesAPI](#).

Inoltre, è possibile prevedere gli eventi di geofence in entrata per un dispositivo che entra, esce o rimane inattivo all'interno di un geofence. Utilizza l'API per prevedere gli eventi.

### [ForecastGeofenceEvents](#)

Se desideri anche tenere traccia della cronologia delle posizioni del dispositivo o visualizzare le posizioni su una mappa, collega il tracker a una raccolta di geofence. In alternativa, potresti non voler valutare tutti gli aggiornamenti sulla posizione o non intendi archiviare i dati sulla posizione in una risorsa di localizzazione. In uno di questi casi, puoi fare una richiesta diretta alla raccolta di geofence e valutare la posizione di uno o più dispositivi rispetto ai geofence.

La valutazione delle posizioni dei dispositivi rispetto ai geofence genera eventi. È possibile reagire a questi eventi e indirizzarli ad altri servizi. AWS Per ulteriori informazioni sulle azioni che puoi intraprendere quando ricevi eventi di geofence, consulta [Reagire agli eventi di Amazon Location Service con Amazon EventBridge](#)

Un evento Amazon Location include gli attributi dell'aggiornamento della posizione del dispositivo che lo genera, inclusi i metadati relativi a ora, posizione, precisione e valore-chiave, e alcuni attributi del geofence da cui si entra o si esce. Per ulteriori informazioni sui dati inclusi in un evento di geofence, consulta [Esempi di EventBridge eventi Amazon per Amazon Location Service](#)

I seguenti esempi utilizzano le AWS CLI, o le API Amazon Location.

## API

Per valutare le posizioni dei dispositivi rispetto alla posizione dei geofence utilizzando le API di Amazon Location

Usa l'[BatchEvaluateGeofences](#) operazione dalle API Amazon Location Geofences.

L'esempio seguente utilizza una richiesta API per valutare la posizione del dispositivo rispetto *ExampleDevice* a una raccolta di geofence associata. *ExampleGeofenceCollection* Sostituisci questi valori con i tuoi ID di geofence e dispositivo.

```
POST /geofencing/v0/collections/ExampleGeofenceCollection/positions HTTP/1.1
Content-type: application/json

{
  "DevicePositionUpdates": [
    {
      "DeviceId": "ExampleDevice",
      "Position": [-123.123, 47.123],
      "SampleTime": "2021-11-30T21:47:25.149Z",
      "Accuracy": {
        "Horizontal": 10.30
      },
      "PositionProperties": {
        "field1": "value1",
        "field2": "value2"
      }
    }
  ]
}
```

## AWS CLI

Per valutare le posizioni dei dispositivi rispetto alla posizione dei geofence utilizzando i comandi AWS CLI

Utilizza il comando [batch-evaluate-geofences](#).

L'esempio seguente utilizza un AWS CLI per valutare la posizione di *ExampleDevice* rispetto a una raccolta di geofence associata. *ExampleGeofenceCollection* Sostituisci questi valori con i tuoi ID di geofence e dispositivo.

```
aws location \
```

```
batch-evaluate-geofences \  
  --collection-name ExampleGeofenceCollection \  
  --device-position-updates '[{"DeviceId": "ExampleDevice", "Position":  
[-123.123, 47.123], "SampleTime": "2021-11-30T21:47:25.149Z", "Accuracy":  
{"Horizontal": 10.30}, "PositionProperties": {"field1": "value1", "field2": "value2"}}]'
```

La valutazione delle posizioni dei dispositivi rispetto ai geofence genera eventi. Tradizionalmente è possibile reagire agli eventi utilizzando [Amazon EventBridge](#), ma questo processo consente di reagire agli eventi solo dopo che si sono verificati. Se devi anticipare quando un dispositivo entra o esce da un geofence, ad esempio se un dispositivo sta attraversando un confine e di conseguenza sarà soggetto a normative diverse, puoi utilizzare l'API [ForecastGeofenceEvents](#) per prevedere eventi di geofence futuri.

L'[ForecastGeofenceEvents](#) API utilizza criteri come il dispositivo, la prossimità time-to-breach, la velocità e la posizione per prevedere gli eventi. L'API restituirà a `ForecastedBreachTime`, che segnala l'ora stimata in cui si verificherà l'evento di geofence.

L'esempio seguente utilizza le API Amazon Location.

## API

Per prevedere gli eventi di geofence utilizzando le API di Amazon Location

Usa l' [ForecastGeofenceEvents](#) operazione dalle API Amazon Location Geofences.

L'esempio seguente utilizza una richiesta API per prevedere gli eventi di geofence per un parente di un. *ExampleDeviceExampleGeofence* Sostituisci questi valori con i tuoi ID di geofence e dispositivo.

```
POST /geofencing/v0/collections/CollectionName/forecast-geofence-events HTTP/1.1  
Content-type: application/json  
  
{  
  "DeviceState": {  
    "Position": [ number ],  
    "Speed": number  
  },  
  "DistanceUnit": "string",  
  "MaxResults": number,  
  "NextToken": "string",  
  "SpeedUnit": "string",
```

```
"TimeHorizonMinutes": number
}
```

## Verifica le posizioni dei dispositivi

Per verificare l'integrità della posizione di un dispositivo, utilizza l'[VerifyDevicePosition](#) API. Questa API restituisce informazioni sull'integrità della posizione del dispositivo, valutando proprietà come il segnale cellulare del dispositivo, il punto di accesso Wi-Fi, l'indirizzo IPv4 e se è in uso un proxy.

### Prerequisiti

Prima di poter utilizzare le API elencate per la verifica del dispositivo, assicurati di disporre dei seguenti prerequisiti:

- Hai creato un tracker per il dispositivo o i dispositivi che desideri controllare. Per ulteriori informazioni, consulta [Inizia il tracciamento](#).

L'esempio seguente mostra una richiesta per l'[VerifyDevicePosition](#) API Amazon Location.

### API

Per verificare le posizioni dei dispositivi utilizzando le API di Amazon Location

Utilizza l'[VerifyDevicePosition](#) operazione delle API di Amazon Location Tracking.

L'esempio seguente mostra una richiesta API per valutare l'integrità della posizione di un dispositivo. Sostituisci questi valori con i tuoi ID di dispositivo.

```
POST /tracking/v0/trackers/TrackerName/positions/verify HTTP/1.1
Content-type: application/json

{
  "DeviceState": {
    "Accuracy": {
      "Horizontal": number
    },
    "CellSignals": {
      "LteCellDetails": [
        {
          "CellId": number,
          "LocalId": {
```

```

        "Earfcn": number,
        "Pci": number
    },
    "Mcc": number,
    "Mnc": number,
    "NetworkMeasurements": [
        {
            "CellId": number,
            "Earfcn": number,
            "Pci": number,
            "Rsrp": number,
            "Rsrq": number
        }
    ],
    "NrCapable": boolean,
    "Rsrp": number,
    "Rsrq": number,
    "Tac": number,
    "TimingAdvance": number
    }
]
},
"DeviceId": "ExampleDevice",
"Ipv4Address": "string",
"Position": [ number ],
"SampleTime": "string",
"WiFiAccessPoints": [
    {
        "MacAddress": "string",
        "Rss": number
    }
]
},
"DistanceUnit": "string"
}

```

### Note

Il Location Integrity SDK offre funzionalità avanzate relative alla verifica dei dispositivi ed è disponibile per l'uso su richiesta. Per accedere all'SDK, contatta il [Sales Support](#).

## Reagire agli eventi di Amazon Location Service con Amazon EventBridge

Amazon EventBridge è un bus di eventi serverless che collega in modo efficiente le applicazioni tra loro utilizzando i dati di AWS servizi come Amazon Location. EventBridge riceve eventi da Amazon Location e indirizza i dati a destinazioni come AWS Lambda. Puoi impostare regole di routing per determinare dove inviare i dati per creare architetture applicative che reagiscono in tempo reale.

Per impostazione predefinita, vengono inviati solo gli eventi di geofence (ENTER e EXIT gli eventi, quando i dispositivi entrano o escono dalle aree geofenzionate). EventBridge Puoi anche abilitare tutti gli eventi di aggiornamento della posizione filtrati per una risorsa tracker. Per ulteriori informazioni, consulta [Abilita gli eventi di aggiornamento per un tracker](#).

Per ulteriori informazioni, consulta [Events and Event Patterns](#) nella Amazon EventBridge User Guide.

### Argomenti

- [Abilita gli eventi di aggiornamento per un tracker](#)
- [Crea regole per eventi per Amazon Location](#)
- [Esempi di EventBridge eventi Amazon per Amazon Location Service](#)

### Abilita gli eventi di aggiornamento per un tracker

Per impostazione predefinita, Amazon Location invia solo eventi di geofence ENTER e EXIT geofence a. EventBridge Puoi abilitare tutti gli UPDATE eventi di posizione filtrati a cui inviare un tracker. EventBridge Puoi farlo quando [crei](#) o [aggiorni](#) un tracker.

Ad esempio, per aggiornare un tracker esistente utilizzando il AWS CLI, puoi usare il seguente comando (usa il nome della tua risorsa tracker al posto di). *MyTracker*

```
aws location update-tracker --tracker-name MyTracker --event-bridge-enabled
```

Per disattivare gli eventi di posizione per un tracker, devi utilizzare l'API o la console Amazon Location Service.

### Crea regole per eventi per Amazon Location

Puoi creare [fino a 300 regole per bus di eventi EventBridge per](#) configurare le azioni intraprese in risposta a un evento Amazon Location.



Ad esempio, puoi creare una regola per gli eventi di geofence in base alla quale verrà inviata una notifica push quando viene rilevato un telefono all'interno di un confine geofenzato.

Per creare una regola per gli eventi di Amazon Location

Utilizzando i seguenti valori, [crea una EventBridge regola](#) basata sugli eventi di Amazon Location:

- Per Rule type (Tipo di regola), scegli Rule with an event pattern (Regola con un modello di eventi).
- Nella casella Schema di evento, aggiungi lo schema seguente:

```
{
  "source": ["aws.geo"],
  "detail-type": ["Location Geofence Event"]
}
```

Per creare una regola per gli aggiornamenti della posizione del tracker, puoi invece utilizzare il seguente schema:

```
{
  "source": ["aws.geo"],
  "detail-type": ["Location Device Position Event"]
}
```

Facoltativamente, puoi specificare solo ENTER EXIT gli eventi aggiungendo un detail tag (se la tua regola riguarda gli aggiornamenti della posizione del tracker, ce n'è solo uno EventType, quindi non è necessario filtrarlo):

```
{
  "source": ["aws.geo"],
  "detail-type": ["Location Geofence Event"],
  "detail": {
    "EventType": ["ENTER"]
  }
}
```

Opzionalmente, puoi anche filtrare in base alle proprietà della posizione o del geofence:

```
{
  "source": ["aws.geo"],
  "detail-type": ["Location Geofence Event"],
```

```
"detail": {
  "EventType": ["ENTER"],
  "GeofenceProperties": {
    "Type": "LoadingDock"
  },
  "PositionProperties": {
    "VehicleType": "Truck"
  }
}
```

- Per gli obiettivi Select, scegli l'azione da intraprendere quando si riceve un evento da Amazon Location Service.

Ad esempio, utilizza un argomento Amazon Simple Notification Service (SNS) per inviare un'e-mail o un messaggio di testo quando si verifica un evento. Per fare ciò, è necessario creare un argomento Amazon SNS utilizzando la console di Amazon SNS. Per ulteriori informazioni, consulta [Using Amazon SNS per le notifiche utente](#).

#### Warning

È buona norma confermare che la regola relativa all'evento sia stata applicata correttamente o che l'azione automatica potrebbe non essere avviata come previsto. Per verificare la regola dell'evento, avvia le condizioni per la regola dell'evento. Ad esempio, simula l'ingresso di un dispositivo in un'area delimitata.

Puoi anche acquisire tutti gli eventi da Amazon Location, semplicemente escludendo la `detail-type` sezione. Per esempio:

```
{
  "source": [
    "aws.geo"
  ]
}
```

**Note**

Lo stesso evento può essere organizzato più di una volta. Puoi utilizzare l'id dell'evento per deduplicare gli eventi che ricevi.

## Esempi di EventBridge eventi Amazon per Amazon Location Service

Di seguito è riportato un esempio di evento per l'accesso a un geofence avviato tramite chiamata `BatchUpdateDevicePosition`

```
{
  "version": "0",
  "id": "aa11aa22-33a-4a4a-aaa5-example",
  "detail-type": "Location Geofence Event",
  "source": "aws.geo",
  "account": "636103698109",
  "time": "2020-11-10T23:43:37Z",
  "region": "eu-west-1",
  "resources": [
    "arn:aws:geo:eu-west-1:0123456789101:geofence-collection/GeofenceEvents-GeofenceCollection_EXAMPLE",
    "arn:aws:geo:eu-west-1:0123456789101:tracker/Tracker_EXAMPLE"
  ],
  "detail": {
    "EventType": "ENTER",
    "GeofenceId": "polygon_14",
    "DeviceId": "Device1-EXAMPLE",
    "SampleTime": "2020-11-10T23:43:37.531Z",
    "Position": [
      -123.12390073297821,
      49.23433613216247
    ],
    "Accuracy": {
      "Horizontal": 15.3
    },
    "GeofenceProperties": {
      "ExampleKey1": "ExampleField1",
      "ExampleKey2": "ExampleField2"
    },
    "PositionProperties": {
      "ExampleKey1": "ExampleField1",
```

```
    "ExampleKey2": "ExampleField2"
  }
}
```

Di seguito è riportato un esempio di evento di uscita da un geofence avviato tramite chiamata `BatchUpdateDevicePosition`

```
{
  "version": "0",
  "id": "aa11aa22-33a-4a4a-aaa5-example",
  "detail-type": "Location Geofence Event",
  "source": "aws.geo",
  "account": "123456789012",
  "time": "2020-11-10T23:41:44Z",
  "region": "eu-west-1",
  "resources": [
    "arn:aws:geo:eu-west-1:0123456789101:geofence-collection/GeofenceEvents-GeofenceCollection_EXAMPLE",
    "arn:aws:geo:eu-west-1:0123456789101:tracker/Tracker_EXAMPLE"
  ],
  "detail": {
    "EventType": "EXIT",
    "GeofenceId": "polygon_10",
    "DeviceId": "Device1-EXAMPLE",
    "SampleTime": "2020-11-10T23:41:43.826Z",
    "Position": [
      -123.08569321875426,
      49.23766166742559
    ],
    "Accuracy": {
      "Horizontal": 15.3
    },
    "GeofenceProperties": {
      "ExampleKey1": "ExampleField1",
      "ExampleKey2": "ExampleField2"
    },
    "PositionProperties": {
      "ExampleKey1": "ExampleField1",
      "ExampleKey2": "ExampleField2"
    }
  }
}
```

Di seguito è riportato un esempio di evento per l'aggiornamento di una posizione, avviato mediante una chiamata. `BatchUpdateDevicePosition`

```
{
  "version": "0",
  "id": "aa11aa22-33a-4a4a-aaa5-example",
  "detail-type": "Location Device Position Event",
  "source": "aws.geo",
  "account": "123456789012",
  "time": "2020-11-10T23:41:44Z",
  "region": "eu-west-1",
  "resources": [
    "arn:aws:geo:eu-west-1:0123456789101:tracker/Tracker_EXAMPLE"
  ],
  "detail": {
    "EventType": "UPDATE",
    "TrackerName": "tracker_2",
    "DeviceId": "Device1-EXAMPLE",
    "SampleTime": "2020-11-10T23:41:43.826Z",
    "ReceivedTime": "2020-11-10T23:41:39.235Z",
    "Position": [
      -123.08569321875426,
      49.23766166742559
    ],
    "Accuracy": {
      "Horizontal": 15.3
    },
    "PositionProperties": {
      "ExampleKey1": "ExampleField1",
      "ExampleKey2": "ExampleField2"
    }
  }
}
```

## Monitoraggio tramite MQTT AWS IoT e Amazon Location Service

[MQTT](#) è un protocollo di messaggistica leggero e ampiamente adottato progettato per dispositivi con limitazioni. AWS IoT Core supporta connessioni di dispositivi che utilizzano il protocollo MQTT e il protocollo MQTT over WebSocket Secure (WSS).

[AWS IoT Core](#) connette i dispositivi AWS e consente di inviare e ricevere messaggi tra di essi. Il motore AWS IoT Core delle regole memorizza le domande sugli argomenti dei messaggi dei tuoi

dispositivi e ti consente di definire azioni per l'invio di messaggi ad altri AWS servizi, come Amazon Location Service. I dispositivi che conoscono la propria posizione come coordinate possono far sì che le proprie posizioni vengano inoltrate ad Amazon Location tramite il motore delle regole.

### Note

I dispositivi possono conoscere la propria posizione, ad esempio tramite il GPS integrato. AWS IoT supporta anche il tracciamento della posizione dei dispositivi di terze parti. Per ulteriori informazioni, consulta [AWS IoT Core Device Location](#) nella AWS IoT Core Developer Guide.

La procedura dettagliata seguente descrive il tracciamento utilizzando AWS IoT Core le regole. Puoi anche inviare le informazioni sul dispositivo alla tua AWS Lambda funzione, se devi elaborarle prima di inviarle ad Amazon Location. Per maggiori dettagli sull'utilizzo di Lambda per elaborare le posizioni dei dispositivi, consulta. [Utilizzo AWS Lambda con MQTT](#)

### Argomenti

- [Prerequisito](#)
- [Creare una regola AWS IoT Core](#)
- [Verifica la tua AWS IoT Core regola nella console](#)
- [Utilizzo AWS Lambda con MQTT](#)

### Prerequisito

Prima di iniziare il tracciamento, devi completare i seguenti prerequisiti:

- [Create una risorsa di tracciamento](#) a cui inviare i dati sulla posizione del dispositivo.
- [Crea un ruolo IAM](#) per concedere AWS IoT Core l'accesso al tuo tracker.

Quando segui questi passaggi, utilizza la seguente politica per consentire l'accesso al tracker:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "WriteDevicePosition",
      "Effect": "Allow",
```

```
    "Action": "geo:BatchUpdateDevicePosition",
    "Resource": "arn:aws:geo:*:*:tracker/*"
  }
]
}
```

## Creare una regola AWS IoT Core

Quindi, crea una AWS IoT Core regola per inoltrare la telemetria posizionale dei tuoi dispositivi ad Amazon Location Service. Per ulteriori informazioni sulla creazione di regole, consulta i seguenti argomenti nella Guida per gli sviluppatori: AWS IoT Core

- [Creazione di una AWS IoT regola](#) per informazioni sulla creazione di una nuova regola.
- [Azione sulla posizione](#) per informazioni specifiche sulla creazione di una regola per la pubblicazione su Amazon Location

## Verifica la tua AWS IoT Core regola nella console

Se al momento nessun dispositivo pubblica dati di telemetria che includono la posizione, puoi testare la regola utilizzando la console. AWS IoT Core La console dispone di un client di test in cui è possibile pubblicare un messaggio di esempio per verificare i risultati della soluzione.

1. Accedi alla AWS IoT Core console all'indirizzo <https://console.aws.amazon.com/iot/>.
2. Nella barra di navigazione a sinistra, espandi Test e scegli MQTT test client.
3. In Pubblica su un argomento, imposta il nome dell'argomento su ***iot/topic*** (o il nome dell'argomento che hai impostato nella AWS IoT Core regola, se diverso) e fornisci quanto segue per il payload del messaggio. Sostituisci il timestamp ***1604940328*** con un timestamp valido degli ultimi 30 giorni (tutti i timestamp più vecchi di 30 giorni vengono ignorati dai tracker di Amazon Location Service).

```
{
  "payload": {
    "deviceid": "thing123",
    "timestamp": 1604940328,
    "location": { "lat": 49.2819, "long": -123.1187 },
    "accuracy": { "Horizontal": 20.5 },
    "positionProperties": { "field1": "value1", "field2": "value2" }
  }
}
```

```
}
```

4. Scegli Pubblica su argomento per inviare il messaggio di prova.
5. Per verificare che il messaggio sia stato ricevuto da Amazon Location Service, usa il seguente AWS CLI comando. Se lo hai modificato durante la configurazione, sostituisci il nome del tracker con quello che hai usato.

```
aws location batch-get-device-position --tracker-name MyTracker --device-ids  
thing123
```

## Utilizzo AWS Lambda con MQTT

Sebbene l'utilizzo non AWS Lambda sia più necessario per l'invio dei dati sulla posizione del dispositivo ad Amazon Location per il tracciamento, in alcuni casi potresti comunque voler utilizzare Lambda. Ad esempio, se desideri elaborare personalmente i dati sulla posizione del tuo dispositivo, prima di inviarli ad Amazon Location. I seguenti argomenti descrivono come utilizzare Lambda per elaborare i messaggi prima di inviarli al tracker. Per ulteriori informazioni su questo modello, consulta l'architettura [di riferimento](#).

### Argomenti

- [Prerequisito](#)
- [Creazione di una funzione Lambda](#)
- [Creare una regola AWS IoT Core](#)
- [Metti alla prova la tua AWS IoT Core regola nella console](#)

### Prerequisito

Prima di iniziare il tracciamento, è necessario [creare una risorsa tracker](#). Per creare una risorsa di tracciamento, puoi utilizzare la console Amazon Location, AWS CLI, o le API Amazon Location.

L'esempio seguente utilizza la console Amazon Location Service per creare la risorsa tracker:

1. Apri la console Amazon Location Service all'[indirizzo https://console.aws.amazon.com/location/](https://console.aws.amazon.com/location/).
2. Nel riquadro di navigazione a sinistra, scegli Trackers.
3. Scegli Crea tracker.
4. Compila le seguenti caselle:



- Nome: inserisci un nome univoco che contenga un massimo di 100 caratteri. Le voci valide includono caratteri alfanumerici, trattini e caratteri di sottolineatura. *MyTracker*Ad esempio,.
  - Descrizione: immettere una descrizione facoltativa. Ad esempio, *Tracker per memorizzare le posizioni dei AWS IoT Core dispositivi*.
  - Filtro della posizione: seleziona il filtro che desideri utilizzare per gli aggiornamenti della posizione. Ad esempio, filtraggio basato sulla precisione.
5. Scegli Crea tracker.

## Creazione di una funzione Lambda

Per creare una connessione tra AWS IoT Core e Amazon Location Service, è necessaria una AWS Lambda funzione per elaborare i messaggi inoltrati da AWS IoT Core. Questa funzione estrarrà tutti i dati di posizione, li formatterà per Amazon Location Service e li invierà tramite l'API Amazon Location Tracker. Puoi creare questa funzione tramite la AWS Lambda console oppure puoi usare AWS Command Line Interface (AWS CLI) o le AWS Lambda API.

Per creare una funzione Lambda che pubblichi aggiornamenti di posizione su Amazon Location utilizzando la console:

1. Apri la console AWS Lambda all'indirizzo <https://console.aws.amazon.com/lambda/>.
2. Dalla barra di navigazione a sinistra, scegli Funzioni.
3. Scegli Crea funzione e assicurati che sia selezionato Author from zero.
4. Compila le seguenti caselle:
  - Nome della funzione: immettere un nome univoco per la funzione. Le voci valide includono caratteri alfanumerici, trattini e caratteri di sottolineatura senza spazi. *MyLambda*Ad esempio,.
  - Runtime — Scegliete *Python 3.8*.
5. Scegli Crea funzione.
6. Scegli la scheda Codice per aprire l'editor.
7. [Sovrascrivi il codice segnaposto `lambda\_function.py` con quanto segue, sostituendo il valore assegnato `TRACKER\_NAME` con il nome del tracker che hai creato come prerequisito.](#)

```
from datetime import datetime
import json
import os
```

```
import boto3

# Update this to match the name of your Tracker resource
TRACKER_NAME = "MyTracker"

"""
This Lambda function receives a payload from AWS IoT Core and publishes device
updates to
Amazon Location Service via the BatchUpdateDevicePosition API.

Parameter 'event' is the payload delivered from AWS IoT Core.

In this sample, we assume that the payload has a single top-level key 'payload' and
a nested key
'location' with keys 'lat' and 'long'. We also assume that the name of the device
is nested in
the payload as 'deviceid'. Finally, the timestamp of the payload is present as
'timestamp'. For
example:

>>> event
{ 'payload': { 'deviceid': 'thing123', 'timestamp': 1604940328,
  'location': { 'lat': 49.2819, 'long': -123.1187 },
  'accuracy': {'Horizontal': 20.5 },
  'positionProperties': {'field1':'value1','field2':'value2'} }
}

If your data doesn't match this schema, you can either use the AWS IoT Core rules
engine to
format the data before delivering it to this Lambda function, or you can modify the
code below to
match it.
"""
def lambda_handler(event, context):
    update = {
        "DeviceId": event["payload"]["deviceid"],
        "SampleTime": datetime.fromtimestamp(event["payload"]
["timestamp"]).strftime("%Y-%m-%dT%H:%M:%SZ"),
        "Position": [
            event["payload"]["location"]["long"],
            event["payload"]["location"]["lat"]
        ]
    }
    if "accuracy" in event["payload"]:
```

```
    update["Accuracy"] = event["payload"]['accuracy']
    if "positionProperties" in event["payload"]:
        update["PositionProperties"] = event["payload"]['positionProperties']

    client = boto3.client("location")
    response = client.batch_update_device_position(TrackerName=TRACKER_NAME,
    Updates=[update])

    return {
        "statusCode": 200,
        "body": json.dumps(response)
    }
```

8. Scegliete Deploy per salvare la funzione aggiornata.
9. Scegli la scheda Configurazione.
10. Nella sezione Autorizzazioni, scegli il nome del ruolo con collegamento ipertestuale per concedere ad Amazon Location Service le autorizzazioni per la tua funzione Lambda.
11. Dalla pagina di riepilogo del tuo ruolo, scegli Aggiungi autorizzazioni, quindi dall'elenco a discesa seleziona Crea politica in linea.
12. Scegli la scheda JSON e sovrascrivi la politica con il seguente documento. Ciò consente alla funzione Lambda di aggiornare le posizioni dei dispositivi gestite da tutte le risorse del tracker in tutte le regioni.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "WriteDevicePosition",
      "Effect": "Allow",
      "Action": "geo:BatchUpdateDevicePosition",
      "Resource": "arn:aws:geo:*:*:tracker/*"
    }
  ]
}
```

13. Scegli Esamina la policy.
14. Inserire un nome per la policy. Ad esempio, *AmazonLocationTrackerWriteOnly*
15. Scegli Crea policy.

È possibile modificare questo codice di funzione, se necessario, per adattarlo allo schema dei messaggi del dispositivo.

## Creare una regola AWS IoT Core

Successivamente, crea una AWS IoT Core regola per inoltrare la telemetria posizionale dei tuoi dispositivi alla AWS Lambda funzione di trasformazione e pubblicazione su Amazon Location Service. La regola di esempio fornita presuppone che qualsiasi trasformazione necessaria del payload del dispositivo venga gestita dalla funzione Lambda. Puoi creare questa regola tramite la AWS IoT Core console, il AWS Command Line Interface (AWS CLI) o le API. AWS IoT Core

### Note

Sebbene la AWS IoT console gestisca le autorizzazioni necessarie AWS IoT Core per consentire di richiamare la funzione Lambda, se stai creando la regola da o SDK, [devi configurare una policy a](#) cui concedere AWS CLI l'autorizzazione. AWS IoT

Per creare un AWS IoT Core file utilizzando la console

1. Accedere alla AWS IoT Core console all'indirizzo <https://console.aws.amazon.com/iot/>.
2. Nella barra di navigazione a sinistra, espandi Act e scegli Rules.
3. Scegli Crea una regola per avviare la nuova procedura guidata per le regole.
4. Immettere un nome e una descrizione per la regola.
5. Per l'istruzione Rule Query, aggiorna l'FROM attributo in modo che faccia riferimento a un argomento in cui almeno un dispositivo pubblica dati di telemetria che include la posizione. Se state testando la soluzione, non è necessaria alcuna modifica.

```
SELECT * FROM 'iot/topic'
```

6. In Imposta una o più azioni, scegli Aggiungi azione.
7. Seleziona Invia un messaggio a una funzione lambda.
8. Selezionare Configure action (Configura operazione).
9. Trova e seleziona la tua funzione Lambda dall'elenco.
10. Selezionare Add action (Aggiungi operazione).
11. Scegli Crea regola.

## Metti alla prova la tua AWS IoT Core regola nella console

Se al momento nessun dispositivo pubblica dati di telemetria che includono la posizione, puoi testare la regola e questa soluzione utilizzando la console. AWS IoT Core La console dispone di un client di test in cui è possibile pubblicare un messaggio di esempio per verificare i risultati della soluzione.

1. Accedi alla AWS IoT Core console all'indirizzo <https://console.aws.amazon.com/iot/>.
2. Nella barra di navigazione a sinistra, espandi Test e scegli MQTT test client.
3. In Pubblica su un argomento, imposta il nome dell'argomento su ***iot/topic*** (o il nome dell'argomento che hai impostato nella AWS IoT Core regola, se diverso) e fornisci quanto segue per il payload del messaggio. Sostituisci il timestamp ***1604940328*** con un timestamp valido degli ultimi 30 giorni (tutti i timestamp più vecchi di 30 giorni vengono ignorati).

```
{
  "payload": {
    "deviceid": "thing123",
    "timestamp": 1604940328,
    "location": { "lat": 49.2819, "long": -123.1187 },
    "accuracy": { "Horizontal": 20.5 },
    "positionProperties": { "field1": "value1", "field2": "value2" }
  }
}
```

4. Scegli Pubblica su argomento per inviare il messaggio di prova.
5. Per verificare che il messaggio sia stato ricevuto da Amazon Location Service, usa il seguente AWS CLI comando. Se li hai modificati durante la configurazione, sostituisci il nome del tracker e l'ID del dispositivo con quelli che hai usato.

```
aws location batch-get-device-position --tracker-name MyTracker --device-ids
thing123
```

## Gestione delle risorse di raccolta di geofence

Gestisci le tue raccolte di geofence utilizzando la console Amazon Location o le AWS CLI API di Amazon Location.

## Elenca le risorse della tua raccolta di geofence

Puoi visualizzare la tua lista di collezioni di geofence utilizzando la console Amazon LocationAWS CLI, o le API di Amazon Location:

### Console

Per visualizzare un elenco di raccolte di geofence utilizzando la console Amazon Location

1. Apri la console Amazon Location all'[indirizzo https://console.aws.amazon.com/location/](https://console.aws.amazon.com/location/).
2. Scegli le collezioni Geofence dal riquadro di navigazione a sinistra.
3. Visualizza un elenco delle tue raccolte di geofence in Le mie collezioni di geofence.

### API

Usa l'[ListGeofenceCollections](#) operazione dalle API Amazon Location Geofences.

L'esempio seguente è una richiesta API per ottenere un elenco di raccolte di geofence nell'account. AWS

```
POST /geofencing/v0/list-collections
```

Di seguito è riportato un esempio di risposta per: ListGeofenceCollections

```
{
  "Entries": [
    {
      "CollectionName": "ExampleCollection",
      "CreateTime": 2020-09-30T22:59:34.142Z,
      "Description": "string",
      "UpdateTime": 2020-09-30T23:59:34.142Z
    },
    "NextToken": "1234-5678-9012"
  ]
}
```

### CLI

Utilizza il comando [list-geofence-collections](#).

L'esempio seguente è un AWS CLI modo per ottenere un elenco di raccolte di geofence presenti nell'AWSaccount.

```
aws location list-geofence-collections
```

## Ottieni i dettagli della collezione di geofence

Puoi ottenere dettagli su qualsiasi risorsa di raccolta di geofence nel tuo AWS account utilizzando la console Amazon LocationAWS CLI, o le API di Amazon Location:

### Console

Per visualizzare i dettagli di una raccolta di geofence utilizzando la console Amazon Location

1. Apri la console Amazon Location all'[indirizzo https://console.aws.amazon.com/location/](https://console.aws.amazon.com/location/).
2. Scegli le collezioni Geofence dal riquadro di navigazione a sinistra.
3. In Le mie collezioni di geofence, seleziona il link del nome della raccolta di geofence di destinazione.

### API

Usa l'[DescribeGeofenceCollection](#) operazione dalle API Amazon Location Geofences.

L'esempio seguente è una richiesta API per cui ottenere i dettagli della raccolta di geofence.

*ExampleCollection*

```
GET /geofencing/v0/collections/ExampleCollection
```

Di seguito è riportato un esempio di risposta per: DescribeGeofenceCollection

```
{
  "CollectionArn": "arn:aws:geo:us-west-2:123456789012:geofence-collection/
GeofenceCollection",
  "CollectionName": "ExampleCollection",
  "CreateTime": 2020-09-30T22:59:34.142Z,
  "Description": "string",
  "KmsKeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
  "Tags": {
    "Tag1" : "Value1"
  },
  "UpdateTime": 2020-09-30T23:59:34.142Z
}
```

## CLI

Utilizza il comando [describe-geofence-collection](#).

L'esempio seguente è un AWS CLI modo per ottenere i dettagli della raccolta di geofence.

*ExampleCollection*

```
aws location describe-geofence-collection \  
  --collection-name "ExampleCollection"
```

## Eliminare una raccolta di geofence

Puoi eliminare una raccolta di geofence dal tuo AWS account utilizzando la console Amazon Location o le API AWS CLI di Amazon Location.

### Console

Per eliminare una raccolta di geofence utilizzando la console Amazon Location

#### Warning

Questa operazione elimina la risorsa in modo permanente.

1. Apri la console Amazon Location all'[indirizzo https://console.aws.amazon.com/location/](https://console.aws.amazon.com/location/).
2. Scegli le collezioni Geofence dal riquadro di navigazione a sinistra.
3. In La mia collezione di geofence, seleziona la raccolta di geofence di destinazione.
4. Scegli Elimina la raccolta di geofence.

### API

Usa l'[DeleteGeofenceCollection](#) operazione dalle API di Amazon Location.

L'esempio seguente è una richiesta API per eliminare la raccolta geofence.

*ExampleCollection*

```
DELETE /geofencing/v0/collections/ExampleCollection
```



Di seguito è riportato un esempio di risposta per: DeleteGeofenceCollection

```
HTTP/1.1 200
```

## CLI

Utilizza il comando [delete-geofence-collection](#).

L'esempio seguente è un AWS CLI comando per eliminare la raccolta geofence.

*ExampleCollection*

```
aws location delete-geofence-collection \  
  --collection-name "ExampleCollection"
```

## Elenca i geofence memorizzati

Puoi elencare i geofence archiviati in una raccolta di geofence specificata utilizzando la console Amazon Location, AWS CLI o le API Amazon Location.

### Console

Per visualizzare un elenco di geofence utilizzando la console Amazon Location

1. Apri la console Amazon Location all'[indirizzo https://console.aws.amazon.com/location/](https://console.aws.amazon.com/location/).
2. Scegli le collezioni Geofence dal riquadro di navigazione a sinistra.
3. In La mia collezione di geofence, seleziona il link del nome della collezione di geofence di destinazione.
4. Visualizza i geofence nella collezione di geofence in Geofences

### API

Usa l'[ListGeofences](#) operazione dalle API Amazon Location Geofences.

L'esempio seguente è una richiesta API per ottenere un elenco di geofence archiviati nella raccolta di geofence. *ExampleCollection*

```
POST /geofencing/v0/collections/ExampleCollection/list-geofences
```

Di seguito è riportato un esempio di risposta per: ListGeofences

```
{
  "Entries": [
    {
      "CreateTime": 2020-09-30T22:59:34.142Z,
      "GeofenceId": "geofence-1",
      "Geometry": {
        "Polygon": [
          [-5.716667, -15.933333,
            [-14.416667, -7.933333],
            [-12.316667, -37.066667],
            [-5.716667, -15.933333]
          ]
        },
      "Status": "ACTIVE",
      "UpdateTime": 2020-09-30T23:59:34.142Z
    }
  ],
  "NextToken": "1234-5678-9012"
}
```

## CLI

Utilizza il comando [list-geofences](#).

L'esempio seguente è un AWS CLI modo per ottenere un elenco di geofence archiviati nella collezione di geofence. *ExampleCollection*

```
aws location list-geofences \
  --collection-name "ExampleCollection"
```

## Ottieni i dettagli del geofence

Puoi ottenere i dettagli di un geofence specifico, come l'ora di creazione, l'ora di aggiornamento, la geometria e lo stato, da una raccolta di geofence utilizzando la console Amazon Location AWS CLI o le API Amazon Location.

## Console

Per visualizzare lo stato di un geofence utilizzando la console Amazon Location

1. Apri la console Amazon Location all'[indirizzo https://console.aws.amazon.com/location/](https://console.aws.amazon.com/location/).
2. Scegli le collezioni Geofence dal riquadro di navigazione a sinistra.
3. In La mia collezione di geofence, seleziona il link del nome della collezione di geofence di destinazione.
4. In Geofences, potrai visualizzare lo stato dei tuoi geofence.

## API

Usa l'[GetGeofence](#) operazione dalle API Amazon Location Geofences.

L'esempio seguente è una richiesta API per ottenere i dettagli del geofence da una raccolta di geofence. *ExampleCollection*

```
GET /geofencing/v0/collections/ExampleCollection/geofences/ExampleGeofence1
```

Di seguito è riportato un esempio di risposta per: GetGeofence

```
{
  "CreateTime": 2020-09-30T22:59:34.142Z,
  "GeofenceId": "ExampleGeofence1",
  "Geometry": {
    "Polygon": [
      [-1,-1],
      [1,-1],
      [0,1],
      [-1,-1]
    ]
  },
  "Status": "ACTIVE",
  "UpdateTime": 2020-09-30T23:59:34.142Z
}
```

## CLI

Utilizza il comando [get-geofence](#).

L'esempio seguente è un AWS CLI modo per ottenere i dettagli della raccolta di geofence. *ExampleCollection*

```
aws location get-geofence \
```

```
--collection-name "ExampleCollection" \  
--geofence-id "ExampleGeofence1"
```

## Eliminare geofences

Puoi eliminare i geofence da una raccolta di geofence utilizzando la console Amazon Location AWS CLI o le API di Amazon Location.

### Console

Per eliminare un geofence utilizzando la console Amazon Location

#### Warning

Questa operazione elimina la risorsa in modo permanente.

1. Apri la console Amazon Location all'[indirizzo https://console.aws.amazon.com/location/](https://console.aws.amazon.com/location/).
2. Scegli le collezioni Geofence dal riquadro di navigazione a sinistra.
3. In La mia collezione di geofence, seleziona il link del nome della collezione di geofence di destinazione.
4. In Geofences, seleziona il geofence di destinazione.
5. Scegli Elimina geofence.

### API

Usa l'[BatchDeleteGeofence](#) operazione dalle API Amazon Location Geofences.

L'esempio seguente è una richiesta API per eliminare geofence dalla raccolta geofence.

*ExampleCollection*

```
POST /geofencing/v0/collections/ExampleCollection/delete-geofences  
Content-type: application/json  
  
{  
  "GeofenceIds": [ "ExampleGeofence11" ]  
}
```

Di seguito è riportato un esempio di risposta riuscita per. [BatchDeleteGeofence](#)

```
HTTP/1.1 200
```

## CLI

Utilizza il comando [batch-delete-geofence](#).

L'esempio seguente è un AWS CLI comando per eliminare i geofence dalla collezione geofence.

*ExampleCollection*

```
aws location batch-delete-geofence \  
  --collection-name "ExampleCollection" \  
  --geofence-ids "ExampleGeofence11"
```

## Gestione delle risorse del tracker

Puoi gestire i tuoi tracker utilizzando la console Amazon LocationAWS CLI, o le API di Amazon Location.

### Elenca i tuoi tracker

Puoi visualizzare il tuo elenco di tracker utilizzando la console Amazon LocationAWS CLI, o le API di Amazon Location:

#### Console

Per visualizzare un elenco di tracker esistenti utilizzando la console Amazon Location

1. Apri la console Amazon Location all'[indirizzo https://console.aws.amazon.com/location/](https://console.aws.amazon.com/location/).
2. Scegli Trackers dalla barra di navigazione a sinistra.
3. Visualizza un elenco delle risorse del tuo tracker in I miei tracker.

#### API

Usa l'[ListTrackers](#)operazione dalle API di Amazon Location Trackers.

L'esempio seguente è una richiesta API per ottenere un elenco di tracker presenti nel tuo account.

AWS

```
POST /tracking/v0/list-trackers
```

Di seguito è riportato un esempio di risposta per: [ListTrackers](#)

```
{
  "Entries": [
    {
      "CreateTime": 2020-10-02T19:09:07.327Z,
      "Description": "string",
      "TrackerName": "ExampleTracker",
      "UpdateTime": 2020-10-02T19:10:07.327Z
    }
  ],
  "NextToken": "1234-5678-9012"
}
```

## CLI

Utilizza il comando [list-trackers](#).

L'esempio seguente serve AWS CLI per ottenere un elenco di tracker presenti nel tuo AWS account.

```
aws location list-trackers
```

## Disconnettere un tracker da una raccolta di geofence

Puoi disconnettere un tracker da una raccolta di geofence utilizzando la console Amazon Location, AWS CLI o le API di Amazon Location:

### Console

Per dissociare un tracker da una raccolta di geofence associata utilizzando la console Amazon Location

1. Apri la console Amazon Location all'[indirizzo https://console.aws.amazon.com/location/](https://console.aws.amazon.com/location/).
2. Scegli Trackers dal riquadro di navigazione a sinistra.
3. In I miei tracker, seleziona il link del nome del tracker di destinazione.

4. In Collezioni di geofence collegate, seleziona una raccolta di geofence con uno stato Collegato.
5. Scegli Unlink.

## API

Usa l'[DisassociateTrackerConsumer](#) operazione dalle API di Amazon Location Trackers.

L'esempio seguente è una richiesta API per dissociare un tracker da una raccolta di geofence associata.

```
DELETE /tracking/v0/trackers/ExampleTracker/consumers/arn:aws:geo:us-west-2:123456789012:geofence-collection/ExampleCollection
```

Di seguito è riportato un esempio di risposta per: [DisassociateTrackerConsumer](#)

```
HTTP/1.1 200
```

## CLI

Utilizza il comando [disassociate-tracker-consumer](#).

L'esempio seguente è un AWS CLI comando per dissociare un tracker da una raccolta di geofence associata.

```
aws location disassociate-tracker-consumer \  
  --consumer-arn "arn:aws:geo:us-west-2:123456789012:geofence-collection/  
ExampleCollection" \  
  --tracker-name "ExampleTracker"
```

## Ottieni i dettagli del tracker

Puoi ottenere dettagli su qualsiasi tracker presente nel tuo AWS account utilizzando la console Amazon Location o le AWS CLI API di Amazon Location.

### Console

Per visualizzare i dettagli del tracker utilizzando la console Amazon Location

1. Apri la console Amazon Location all'[indirizzo https://console.aws.amazon.com/location/](https://console.aws.amazon.com/location/).
2. Scegli Trackers dalla barra di navigazione a sinistra.
3. In I miei tracker, seleziona il link del nome del tracker di destinazione.
4. Visualizza i dettagli del tracker in Informazioni.

## API

Usa l'[DescribeTracker](#) operazione dalle API di Amazon Location Tracker.

L'esempio seguente è una richiesta API per cui ottenere i dettagli del tracker. *ExampleTracker*

```
GET /tracking/v0/trackers/ExampleTracker
```

Di seguito è riportato un esempio di risposta per [DescribeTracker](#):

```
{
  "CreateTime": 2020-10-02T19:09:07.327Z,
  "Description": "string",
  "EventBridgeEnabled": false,
  "KmsKeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
  "PositionFiltering": "TimeBased",
  "Tags": {
    "Tag1" : "Value1"
  },
  "TrackerArn": "arn:aws:geo:us-west-2:123456789012:tracker/ExampleTracker",
  "TrackerName": "ExampleTracker",
  "UpdateTime": 2020-10-02T19:10:07.327Z
}
```

## CLI

Utilizza il comando [describe-tracker](#).

L'esempio seguente è un AWS CLI comando per cui ottenere i dettagli del tracker.

*ExampleTracker*

```
aws location describe-tracker \  
  --tracker-name "ExampleTracker"
```



## Eliminare un tracker

Puoi eliminare un tracker dal tuo AWS account utilizzando la console Amazon LocationAWS CLI, o le API Amazon Location:

### Console

Per eliminare una risorsa cartografica esistente utilizzando la console Amazon Location

#### Warning

Questa operazione elimina la risorsa in modo permanente. Se la risorsa tracker è in uso, è possibile che si verifichi un errore. Assicurati che la risorsa di destinazione non sia una dipendenza per le tue applicazioni.

1. Apri la console Amazon Location all'[indirizzo https://console.aws.amazon.com/location/](https://console.aws.amazon.com/location/).
2. Scegli Trackers dal riquadro di navigazione a sinistra.
3. In I miei tracker, seleziona il tracker di destinazione.
4. Scegli Elimina tracker.

### API

Usa l'[DeleteTracker](#)operazione dalle API di Amazon Location Tracker.

L'esempio seguente è una richiesta API per eliminare il tracker. *ExampleTracker*

```
DELETE /tracking/v0/trackers/ExampleTracker
```

Di seguito è riportato un esempio di risposta per[DeleteTracker](#):

```
HTTP/1.1 200
```

### CLI

Utilizza il comando [delete-tracker](#).

L'esempio seguente è un AWS CLI comando per eliminare il tracker *ExampleTracker*.

```
aws location delete-tracker \
```

```
--tracker-name "ExampleTracker"
```

## Esempio di applicazione mobile Geofencing and Tracking

Questo argomento tratta dei tutorial progettati per dimostrare le caratteristiche principali dell'uso dei geofence e dei tracker di Amazon Location in un'applicazione mobile. Le applicazioni dimostrano come interagiscono un tracker e un geofence utilizzando una combinazione di funzionalità Lambda AWS IoT e Amazon Location. Sono disponibili due tutorial.

- [Esempio di applicazione di tracciamento e geofencing per Android e puoi clonare i file di progetto da: https://github.com/aws-geospatial/ /tree/main/](https://github.com/aws-geospatial/tree/main/). [GitHub amazon-location-samples-android tracking-with-geofence-notifications](#)
- [Esempio di applicazione di tracciamento e geofencing per iOS e puoi clonare i file di progetto da GitHub: https://github.com/aws-geospatial/ /tree/main/](https://github.com/aws-geospatial/tree/main/). [amazon-location-samples-ios tracking-with-geofence-notifications](#)

## Esempio di applicazione di tracciamento e geofence per Android

Questo argomento tratta il tutorial per Android progettato per dimostrare le caratteristiche principali dell'uso dei geofence e dei tracker di Amazon Location in un'applicazione mobile. Le applicazioni dimostrano come interagiscono un tracker e un geofence utilizzando una combinazione di funzionalità Lambda AWS IoT e Amazon Location.

### Argomenti

- [Crea risorse Amazon Location per la tua app](#)
- [Crea una collezione Geofence](#)
- [Collega un tracker a una raccolta di geofence](#)
- [Usare AWS Lambda con MQTT](#)
- [Configura il codice dell'app di esempio](#)
- [Utilizzando l'app di esempio](#)

### Crea risorse Amazon Location per la tua app

Per iniziare, dovrai creare le risorse Amazon Location richieste. Queste risorse saranno essenziali per la funzionalità dell'applicazione e l'esecuzione dei frammenti di codice forniti.

**Note**

Se non hai ancora creato un AWS account, segui le istruzioni nella guida per l'utente di [gestione AWS dell'account](#).

Per iniziare devi creare un ID del pool di identità di Amazon Cognito, utilizza la seguente procedura:

1. Apri la [console Amazon Cognito](#) e seleziona Pool di identità dal menu a sinistra, quindi seleziona Crea pool di identità.
2. Assicurati che Guest Access sia selezionato e premi Avanti per continuare.
3. Quindi crea un nuovo ruolo IAM o utilizza un ruolo IAM esistente.
4. Inserisci il nome di un pool di identità e assicurati che Identity Pool abbia accesso alle (geo) risorse di Amazon Location per la mappa e il tracker che creerai nella procedura successiva.

Successivamente devi creare e definire lo stile di una mappa nella console AWS Amazon Location, utilizza la seguente procedura:

1. Vai alla [sezione Mappe](#) della console Amazon Location e seleziona Crea mappa.
2. Assegna un nome e una descrizione alla nuova mappa. Registra il nome che assegni, così come verrà usato più avanti nel tutorial.
3. Quando scegli uno stile di mappa, considera il fornitore dei dati cartografici. Per maggiori dettagli, consulta la sezione 82 dei [termini del AWS servizio](#).
4. Accetta i [Termini e condizioni di Amazon Location](#), quindi seleziona Crea mappa per completare il processo di creazione della mappa.

Successivamente devi creare un tracker nella console Amazon Location, utilizza la seguente procedura:

1. Apri la [sezione Mappe](#) nella console Amazon Location.
2. Scegli Crea tracker.
3. Compila i campi obbligatori. Prendi nota del nome del tracker poiché verrà utilizzato come riferimento nel corso di questo tutorial.

4. Nel campo Filtro della posizione, scegli l'opzione più adatta a come intendi utilizzare la risorsa del tracker. Se non imposti il filtro di posizione, l'impostazione predefinita è TimeBased Per ulteriori informazioni, consulta [Trackers](#) e [PositionFiltering](#) nell'Amazon Location API Reference.
5. Scegli Crea tracker per completare la creazione del tracker.

## Crea una collezione Geofence

Ora creerai una collezione di geofence. Puoi utilizzare la console, l'API o la CLI. Le seguenti procedure illustrano ciascuna opzione.

- Crea una raccolta di geofence utilizzando la console Amazon Location:
  1. Apri la sezione [Geofence Collections](#) della console Amazon Location.
  2. Scegli Crea una collezione di geofence.
  3. Fornisci un nome e una descrizione per la raccolta.
  4. In base alla EventBridge regola con Amazon CloudWatch come obiettivo, puoi creare una EventBridge regola opzionale per iniziare a reagire agli eventi di geofence. Ciò consente ad Amazon Location di pubblicare eventi su Amazon CloudWatch Logs.
  5. Premi la raccolta Create geofence per completare la creazione della raccolta.
- Crea una raccolta di geofence utilizzando l'API Amazon Location:

Usa l'[CreateGeofenceCollection](#) operazione dalle API Amazon Location Geofences.

L'esempio seguente utilizza una richiesta API per creare una raccolta di geofence chiamata.

***GEOCOLLECTION\_NAME***

```
POST /geofencing/v0/collections
Content-type: application/json

{
  "CollectionName": "GEOCOLLECTION_NAME",
  "Description": "Geofence collection 1 for shopping center",
  "Tags": {
    "Tag1" : "Value1"
  }
}
```

- Crea una raccolta di geofence utilizzando i comandi CLI: AWS

Utilizza il comando `create-geofence-collection`. L'esempio seguente utilizza una AWS CLI per creare una raccolta di geofence chiamata. *GEOCOLLECTION\_NAME* Per ulteriori informazioni sull'uso della AWS CLI, consulta la documentazione dell'[interfaccia a riga di AWS comando](#).

```
aws location \  
  create-geofence-collection \  
  --collection-name "ExampleGeofenceCollection" \  
  --description "Shopping center geofence collection" \  
  --tags Tag1=Value1
```

## Collega un tracker a una raccolta di geofence

Per collegare un tracker a una raccolta di geofence puoi utilizzare la console, l'API o la CLI. Le seguenti procedure illustrano ciascuna opzione.

Collega una risorsa tracker a una raccolta di geofence utilizzando la console Amazon Location Service:

1. Apri la console Amazon Location.
2. Nel riquadro di navigazione a sinistra, scegli Trackers.
3. In Device Trackers, seleziona il link del nome del tracker di destinazione.
4. In Collezioni Geofence collegate, scegli Collega collezione Geofence.
5. Nella finestra Linked Geofence Collection, seleziona una collezione di geofence dal menu a discesa.
6. Scegliere Link (Collegamento).
7. Dopo aver collegato la risorsa tracker, ad essa verrà assegnato uno stato Attivo.

Collega una risorsa tracker a una raccolta di geofence utilizzando le API di localizzazione di Amazon:

Usa l'`AssociateTrackerConsumer` operazione dalle API di Amazon Location Trackers. L'esempio seguente utilizza una richiesta API che associa un a una `ExampleTracker` raccolta di geofence utilizzando il relativo Amazon Resource Name (ARN).

```
POST /tracking/v0/trackers/ExampleTracker/consumers  
Content-type: application/json  
{
```

```
"ConsumerArn": "arn:aws:geo:us-west-2:123456789012:geofence-
collection/GOECOLLECTION_NAME"
}
```

Collega una risorsa tracker a una raccolta di geofence utilizzando i comandi CLI AWS :

Utilizza il comando `associate-tracker-consumer` . L'esempio seguente utilizza una AWS CLI per creare una raccolta di geofence chiamata. *GOECOLLECTION\_NAME*

```
aws location \
associate-tracker-consumer \
  --consumer-arn "arn:aws:geo:us-west-2:123456789012:geofence-
collection/GOECOLLECTION_NAME" \
  --tracker-name "ExampleTracker"
```

## Usare AWS Lambda con MQTT

Per creare una connessione tra Amazon Location AWS IoT e Amazon, è necessaria una funzione Lambda per elaborare i messaggi inoltrati dagli eventi. EventBridge CloudWatch Questa funzione estrarrà tutti i dati di posizione, li formatterà per Amazon Location e li invierà tramite l'API Amazon Location Tracker.

La procedura seguente mostra come creare questa funzione tramite la console Lambda:

1. Apri la [console](#).
2. Dalla barra di navigazione a sinistra, scegli Funzioni.
3. Quindi scegli Crea funzione e assicurati che sia selezionata l'opzione Autore da zero.
4. fornite un nome per la funzione e, per l'opzione Runtime, scegliete Node.js 16.x.
5. Scegli Crea funzione.
6. Aprite la scheda Codice per accedere all'editor.
7. Sovrascrivi il codice segnaposto nel `index.js` file con quanto segue:

```
const AWS = require('aws-sdk')
const iot = new AWS.Iot();
exports.handler = function(event) {
  console.log("event===>>>", JSON.stringify(event));
```

```

var param = {
  endpointType: "iot:Data-ATS"
};
iot.describeEndpoint(param, function(err, data) {
  if (err) {
    console.log("error===>>>", err, err.stack); // an error occurred
  } else {
    var endp = data['endpointAddress'];
    const iotdata = new AWS.IotData({endpoint: endp});
    const trackerEvent = event["detail"]["EventType"];
    const src = event["source"];
    const time = event["time"];
    const gfId = event["detail"]["GeofenceId"];
    const resources = event["resources"][0];
    const splitResources = resources.split(".");
    const geofenceCollection = splitResources[splitResources.length -
1];

    const coordinates = event["detail"]["Position"];

    const deviceId = event["detail"]["DeviceId"];
    console.log("deviceId===>>>", deviceId);
    const msg = {
      "trackerEventType" : trackerEvent,
      "source" : src,
      "eventTime" : time,
      "geofenceId" : gfId,
      "coordinates": coordinates,
      "geofenceCollection": geofenceCollection
    };
    const params = {
      topic: `${deviceId}/tracker`,
      payload: JSON.stringify(msg),
      qos: 0
    };
    iotdata.publish(params, function(err, data) {
      if (err) {
        console.log("error===>>>", err, err.stack); // an error
occurred

      } else {
        console.log("Ladmbda triggered===>>>", trackerEvent); //
successful response
      }
    });
  }
});
}

```

```
});  
}
```

8. Premete Deploy per salvare la funzione aggiornata.
9. Quindi apri la scheda Configurazione.
10. Nella sezione Trigger, premi il pulsante Aggiungi trigger.
11. Seleziona EventBridge (CloudWatch Eventi) nel campo Origine.
12. Seleziona l'opzione Regole esistenti.
13. Inserisci il nome della regola, ad esempio `AmazonLocationMonitor-GEOFENCECOLLECTION_NAME`.
14. Premi il pulsante Aggiungi.
15. Verranno inoltre allegate dichiarazioni politiche basate sulle risorse nella scheda delle autorizzazioni

Ora configurerai il MQTT Test Client utilizzando AWS IoT, usa la seguente procedura:

1. Aprire <https://console.aws.amazon.com/iot/>.
2. Nel riquadro di navigazione a sinistra, selezionate il client di test MQTT.
3. Verrà visualizzata una sezione intitolata Client di test MQTT in cui è possibile configurare la connessione MQTT.
4. Dopo aver configurato le impostazioni necessarie, fare clic sul pulsante Connect per stabilire una connessione al broker MQTT utilizzando i parametri forniti.
5. Registra l'endpoint, così come verrà utilizzato più avanti nel tutorial.

Una volta connessi al client di test, è possibile abbonarsi agli argomenti MQTT o pubblicare messaggi sugli argomenti utilizzando i rispettivi campi di input forniti nell'interfaccia del client di test MQTT. Successivamente creerai una AWS IoT politica.

6. Nel menu a sinistra, in Gestione, espandi l'opzione Sicurezza e fai clic su Politiche.
7. Fai clic sul pulsante Crea politica.
8. Inserire un nome per la policy.
9. In Policy Document seleziona la scheda JSON.
10. Copia e incolla la politica mostrata di seguito, ma assicurati di aggiornare tutti gli elementi con il tuo **REGION** e **ACCOUNT\_ID**:



```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "iot:Connect",
        "iot:Publish",
        "iot:Subscribe",
        "iot:Receive"
      ],
      "Resource": [
        "arn:aws:iot:REGION:ACCOUNT_ID:client/${cognito-identity.amazonaws.com:sub}",
        "arn:aws:iot:REGION:ACCOUNT_ID:topic/${cognito-identity.amazonaws.com:sub}",
        "arn:aws:iot:REGION:ACCOUNT_ID:topicfilter/${cognito-identity.amazonaws.com:sub}/*",
        "arn:aws:iot:REGION:ACCOUNT_ID:topic/${cognito-identity.amazonaws.com:sub}/tracker"
      ],
      "Effect": "Allow"
    }
  ]
}
```

11. Seleziona il pulsante Crea per terminare.

Dopo aver completato la procedura precedente, ora aggiornerai le autorizzazioni per il ruolo di ospite come segue:

1. Accedi ad Amazon Cognito e apri il tuo pool di identità. Quindi, procedi all'accesso utente e seleziona il ruolo di ospite.
2. Fai clic sulle politiche di autorizzazione per abilitare la modifica.

```
{
  'Version': '2012-10-17',
  'Statement': [
    {
      'Action': [
        'geo:GetMap*',
        'geo:BatchUpdateDevicePosition',

```

```

        'geo:BatchEvaluateGeofences',
        'iot:Subscribe',
        'iot:Publish',
        'iot:Connect',
        'iot:Receive',
        'iot:AttachPrincipalPolicy',
        'iot:AttachPolicy',
        'iot:DetachPrincipalPolicy',
        'iot:DetachPolicy'
    ],
    'Resource': [
        'arn:aws:geo:us-east-1:{USER_ID}:map/{MAP_NAME}',
        'arn:aws:geo:us-east-1:{USER_ID}:tracker/{TRACKER_NAME}',
        'arn:aws:geo:us-east-1:{USER_ID}:geofence-collection/
{GEOFENCE_COLLECTION_NAME}',
        'arn:aws:iot:us-east-1:{USER_ID}:client/${cognito-
identity.amazonaws.com:sub}',
        'arn:aws:iot:us-east-1:{USER_ID}:topic/${cognito-
identity.amazonaws.com:sub}',
        'arn:aws:iot:us-east-1:{USER_ID}:topicfilter/${cognito-
identity.amazonaws.com:sub}/*',
        'arn:aws:iot:us-east-1:{USER_ID}:topic/${cognito-
identity.amazonaws.com:sub}/tracker'
    ],
    'Effect': 'Allow'
},
{
    'Condition': {
        'StringEquals': {
            'cognito-identity.amazonaws.com:sub': '${cognito-
identity.amazonaws.com:sub}'
        }
    },
    'Action': [
        'iot:AttachPolicy',
        'iot:DetachPolicy',
        'iot:AttachPrincipalPolicy',
        'iot:DetachPrincipalPolicy'
    ],
    'Resource': [
        '*'
    ],
    'Effect': 'Allow'
}

```

```
]
}
```

3. Con le modifiche alle policy di cui sopra, tutte le AWS risorse necessarie sono ora configurate in modo appropriato per l'applicazione.

Configura il codice dell'app di esempio

1. Clona questo repository: <https://github.com/aws-geospatial/amazon-location-samples-android-tracking-with-geofence-notifications/tree/main/> sul tuo computer locale.
2. Apri il progetto in Android Studio. AmazonSampleSDKApp
3. Crea ed esegui l'app sul tuo dispositivo o emulatore Android.

Utilizzando l'app di esempio

Per utilizzare l'esempio, segui queste procedure:

- Crea un **custom.properties**:

Per configurare il `custom.properties` file, procedi nel seguente modo:

1. Apri il tuo editor di testo o IDE preferito.
2. Crea un nuovo file.
3. Salva il file con il nome `custom.properties`.
4. Aggiorna il file `custom.properties` con il seguente esempio di codice e sostituisci `MQTT_END_POINT`, `POLICY_NAME`, `GEOFENCE_COLLECTION_NAME`, e `TOPIC_TRACKER` con i nomi delle tue risorse:

```
MQTT_END_POINT=YOUR_END_POINT.us-east-1.amazonaws.com
POLICY_NAME=YOUR_POLICY
GEOFENCE_COLLECTION_NAME=YOUR_GEOFENCE
TOPIC_TRACKER=YOUR_TRACKER
```

5. Pulisci e ricostruisci il progetto. Dopodiché, puoi eseguire il progetto.
- Accedi:

Per accedere all'applicazione, procedi nel seguente modo:

1. Premi il pulsante Accedi.
  2. Fornisci un ID del pool di identità, un nome del tracker e un nome della mappa.
  3. Premi nuovamente Accedi per terminare.
- Gestisci filtri:

Apri la schermata di configurazione ed esegui le seguenti operazioni:

1. Attiva o disattiva i filtri utilizzando l'interfaccia utente dello switch.
  2. Aggiorna i filtri di tempo e distanza quando necessario.
- Operazioni di tracciamento:

Apri la schermata di tracciamento ed esegui le seguenti operazioni:

- È possibile avviare e interrompere il tracciamento in primo piano, sullo sfondo o in modalità risparmio energetico premendo i rispettivi pulsanti.

## Esempio di applicazione di tracciamento e geofencing per iOS

Questo argomento tratta il tutorial iOS progettato per dimostrare le caratteristiche principali dell'uso dei geofence e dei tracker di Amazon Location in un'applicazione mobile. Le applicazioni dimostrano come interagiscono un tracker e un geofence utilizzando una combinazione di funzionalità Lambda AWS IoT e Amazon Location.

### Argomenti

- [Crea risorse Amazon Location per la tua app](#)
- [Crea una collezione Geofence](#)
- [Collega un tracker a una raccolta di geofence](#)
- [Usare AWS Lambda con MQTT](#)
- [Configurazione di un codice di esempio per l'app](#)
- [Utilizzando l'app di esempio](#)

### Crea risorse Amazon Location per la tua app

Per iniziare, dovrai creare le risorse Amazon Location richieste. Queste risorse saranno essenziali per la funzionalità dell'applicazione e l'esecuzione dei frammenti di codice forniti.

**Note**

Se non hai ancora creato un AWS account, segui le istruzioni nella guida per l'utente di [gestione AWS dell'account](#).

Per iniziare devi creare un ID del pool di identità di Amazon Cognito, utilizza la seguente procedura:

1. Apri la [console Amazon Cognito](#) e seleziona Pool di identità dal menu a sinistra, quindi seleziona Crea pool di identità.
2. Assicurati che Guest Access sia selezionato e premi Avanti per continuare.
3. Quindi crea un nuovo ruolo IAM o utilizza un ruolo IAM esistente.
4. Inserisci il nome di un pool di identità e assicurati che Identity Pool abbia accesso alle (geo) risorse di Amazon Location per la mappa e il tracker che creerai nella procedura successiva.

Successivamente devi creare e definire lo stile di una mappa nella console AWS Amazon Location, utilizza la seguente procedura:

1. Vai alla [sezione Mappe](#) della console Amazon Location e seleziona Crea mappa.
2. Assegna un nome e una descrizione alla nuova mappa. Registra il nome che assegna, così come verrà usato più avanti nel tutorial.
3. Quando scegli uno stile di mappa, considera il fornitore dei dati cartografici. Per maggiori dettagli, consulta la sezione 82 dei [termini del AWS servizio](#).
4. Accetta i [Termini e condizioni di Amazon Location](#), quindi seleziona Crea mappa per completare il processo di creazione della mappa.

Successivamente devi creare un tracker nella console Amazon Location, utilizza la seguente procedura:

1. Apri la [sezione Mappe](#) nella console Amazon Location.
2. Scegli Crea tracker.
3. Compila i campi obbligatori. Prendi nota del nome del tracker poiché verrà utilizzato come riferimento nel corso di questo tutorial.
4. Nel campo Filtro della posizione, scegli l'opzione più adatta a come intendi utilizzare la risorsa del tracker. Se non imposti il filtro di posizione, l'impostazione predefinita è TimeBased Per

ulteriori informazioni, consulta [Start tracking](#) e [PositionFiltering](#) nell'Amazon Location API Reference.

5. Scegli Crea tracker per completare la creazione del tracker.

## Crea una collezione Geofence

Ora creerai una collezione di geofence. Puoi utilizzare la console, l'API o la CLI. Le seguenti procedure illustrano ciascuna opzione.

- Crea una raccolta di geofence utilizzando la console Amazon Location:
  1. Apri la sezione [Geofence Collections](#) della console Amazon Location.
  2. Scegli Crea una collezione di geofence.
  3. Fornisci un nome e una descrizione per la raccolta.
  4. In base alla EventBridge regola con Amazon CloudWatch come obiettivo, puoi creare una EventBridge regola opzionale per iniziare a reagire agli eventi di geofence. Ciò consente ad Amazon Location di pubblicare eventi su Amazon CloudWatch Logs.
  5. Premi la raccolta Create geofence per completare la creazione della raccolta.
- Crea una raccolta di geofence utilizzando l'API Amazon Location:

Usa l'[CreateGeofenceCollection](#) operazione dalle API Amazon Location Geofences.

L'esempio seguente utilizza una richiesta API per creare una raccolta di geofence chiamata.

***GEOCOLLECTION\_NAME***

```
POST /geofencing/v0/collections
Content-type: application/json

{
  "CollectionName": "GEOCOLLECTION_NAME",
  "Description": "Geofence collection 1 for shopping center",
  "Tags": {
    "Tag1" : "Value1"
  }
}
```

- Crea una raccolta di geofence utilizzando i comandi CLI: AWS

Utilizza il comando `create-geofence-collection`. L'esempio seguente utilizza una AWS CLI per creare una raccolta di geofence chiamata `GEOCOLLECTION_NAME`. Per ulteriori informazioni sull'uso della AWS CLI, consulta la documentazione dell'[interfaccia a riga di AWS comando](#).

```
aws location \  
  create-geofence-collection \  
  --collection-name "ExampleGeofenceCollection" \  
  --description "Shopping center geofence collection" \  
  --tags Tag1=Value1
```

## Collega un tracker a una raccolta di geofence

Per collegare un tracker a una raccolta di geofence puoi utilizzare la console, l'API o la CLI. Le seguenti procedure illustrano ciascuna opzione.

Collega una risorsa tracker a una raccolta di geofence utilizzando la console Amazon Location Service:

1. Apri la console Amazon Location.
2. Nel riquadro di navigazione a sinistra, scegli Trackers.
3. In Device Trackers, seleziona il link del nome del tracker di destinazione.
4. In Collezioni Geofence collegate, scegli Collega collezione Geofence.
5. Nella finestra Linked Geofence Collection, seleziona una collezione di geofence dal menu a discesa.
6. Scegliere Link (Collegamento).
7. Dopo aver collegato la risorsa tracker, ad essa verrà assegnato uno stato Attivo.

Collega una risorsa tracker a una raccolta di geofence utilizzando le API di localizzazione di Amazon:

Usa l'AssociateTrackerConsumeroperazione dalle API di Amazon Location Trackers. L'esempio seguente utilizza una richiesta API associata ExampleTracker a una raccolta di geofence utilizzando il relativo Amazon Resource Name (ARN).

```
POST /tracking/v0/trackers/ExampleTracker/consumers  
Content-type: application/json  
{
```

```
"ConsumerArn": "arn:aws:geo:us-west-2:123456789012:geofence-
collection/GEOCOLLECTION_NAME"
}
```

Collega una risorsa tracker a una raccolta di geofence utilizzando i comandi CLI di AWS:

Utilizza il comando `associate-tracker-consumer` . L'esempio seguente utilizza una CLI AWS per creare una raccolta di geofence chiamata. *GEOCOLLECTION\_NAME*

```
aws location \
  associate-tracker-consumer \
    --consumer-arn "arn:aws:geo:us-west-2:123456789012:geofence-
collection/GEOCOLLECTION_NAME" \
    --tracker-name "ExampleTracker"
```

## Usare AWS Lambda con MQTT

Per creare una connessione tra Amazon Location AWS IoT e Amazon, è necessaria una funzione Lambda per elaborare i messaggi inoltrati dagli eventi. EventBridge CloudWatch Questa funzione estrarrà tutti i dati di posizione, li formatterà per Amazon Location e li invierà tramite l'API Amazon Location Tracker.

La procedura seguente mostra come creare questa funzione tramite la console Lambda:

1. Apri la [console](#).
2. Dalla barra di navigazione a sinistra, scegli Funzioni.
3. Quindi scegli Crea funzione e assicurati che sia selezionata l'opzione Autore da zero.
4. fornite un nome per la funzione e, per l'opzione Runtime, scegliete Node.js 16.x.
5. Scegli Crea funzione.
6. Aprite la scheda Codice per accedere all'editor.
7. Sovrascrivi il codice segnaposto nel `index.js` file con quanto segue:

```
const AWS = require('aws-sdk')
const iot = new AWS.Iot();
exports.handler = function(event) {
  console.log("event===>>>", JSON.stringify(event));
  var param = {
    endpointType: "iot:Data-ATS"
  };
};
```



```

iot.describeEndpoint(param, function(err, data) {
  if (err) {
    console.log("error===>>>", err, err.stack); // an error occurred
  } else {
    var endp = data['endpointAddress'];
    const iotdata = new AWS.IotData({endpoint: endp});
    const trackerEvent = event["detail"]["EventType"];
    const src = event["source"];
    const time = event["time"];
    const gfId = event["detail"]["GeofenceId"];
    const resources = event["resources"][0];
    const splitResources = resources.split(".");
    const geofenceCollection = splitResources[splitResources.length -
1];

    const coordinates = event["detail"]["Position"];

    const deviceId = event["detail"]["DeviceId"];
    console.log("deviceId===>>>", deviceId);
    const msg = {
      "trackerEventType" : trackerEvent,
      "source" : src,
      "eventTime" : time,
      "geofenceId" : gfId,
      "coordinates": coordinates,
      "geofenceCollection": geofenceCollection
    };
    const params = {
      topic: `${deviceId}/tracker`,
      payload: JSON.stringify(msg),
      qos: 0
    };
    iotdata.publish(params, function(err, data) {
      if (err) {
        console.log("error===>>>", err, err.stack); // an error
occurred
      } else {
        console.log("Ladmbda triggered===>>>", trackerEvent); //
successful response
      }
    });
  }
});
}

```

8. Premete Deploy per salvare la funzione aggiornata.
9. Quindi apri la scheda Configurazione.
10. Nella sezione Trigger, premi il pulsante Aggiungi trigger.
11. Seleziona EventBridge (CloudWatch Eventi) nel campo Origine.
12. Seleziona l'opzione Regole esistenti.
13. Inserisci il nome della regola, ad esempio AmazonLocationMonitor-GEOFENCECOLLECTION\_NAME.
14. Premi il pulsante Aggiungi.
15. Verranno inoltre allegate dichiarazioni politiche basate sulle risorse nella scheda delle autorizzazioni

Ora configurerai il AWS IoT MQTT Test Client, usa la seguente procedura:

1. Apri <https://console.aws.amazon.com/iot/>.
2. Nel riquadro di navigazione a sinistra, selezionate il client di test MQTT.
3. Verrà visualizzata una sezione intitolata Client di test MQTT in cui è possibile configurare la connessione MQTT.
4. Dopo aver configurato le impostazioni necessarie, fare clic sul pulsante Connect per stabilire una connessione al broker MQTT utilizzando i parametri forniti.
5. Registra l'endpoint, così come verrà utilizzato più avanti nel tutorial.

Una volta connessi al client di test, è possibile abbonarsi agli argomenti MQTT o pubblicare messaggi sugli argomenti utilizzando i rispettivi campi di input forniti nell'interfaccia del client di test MQTT. Successivamente creerai una AWS IoT politica.

6. Nel menu a sinistra, in Gestione, espandi l'opzione Sicurezza e fai clic su Politiche.
7. Fai clic sul pulsante Crea politica.
8. Inserire un nome per la policy.
9. In Policy Document seleziona la scheda JSON.
10. Copia e incolla la politica mostrata di seguito, ma assicurati di aggiornare tutti gli elementi con il tuo **REGION** e **ACCOUNT\_ID**:

```
{
  "Version": "2012-10-17",
  "Statement": [
```

```

{
  "Action": [
    "iot:Connect",
    "iot:Publish",
    "iot:Subscribe",
    "iot:Receive"
  ],
  "Resource": [
    "arn:aws:iot:REGION:ACCOUNT_ID:client/${cognito-identity.amazonaws.com:sub}",
    "arn:aws:iot:REGION:ACCOUNT_ID:topic/${cognito-identity.amazonaws.com:sub}",
    "arn:aws:iot:REGION:ACCOUNT_ID:topicfilter/${cognito-identity.amazonaws.com:sub}/*",
    "arn:aws:iot:REGION:ACCOUNT_ID:topic/${cognito-identity.amazonaws.com:sub}/tracker"
  ],
  "Effect": "Allow"
}
]
}

```

11. Seleziona il pulsante Crea per terminare.

## Configurazione di un codice di esempio per l'app

Per configurare il codice di esempio è necessario che siano installati i seguenti strumenti:

- Git
- XCode 15.3 o versione successiva
- iOS Simulator 16 o versioni successive

Usa questa procedura per configurare il codice dell'app di esempio:

1. Clona il repository git da questo URL: <https://github.com/aws-geospatial/amazon-location-samples-ios/tree/main/tracking-with-geofence-notifications>
2. Apri il file di progetto `AWSLocationSampleApp.xcodeproj`.
3. Attendi il completamento del processo di risoluzione dei pacchetti.
4. Nel menu di navigazione del progetto, `ConfigTemplate.xcconfig` rinominate `Config.xcconfig` e inserite i seguenti valori:

```
IDENTITY_POOL_ID = `YOUR_IDENTITY_POOL_ID`  
MAP_NAME = `YOUR_MAP_NAME`  
TRACKER_NAME = `YOUR_TRACKER_NAME`  
WEBSOCKET_URL = `YOUR_MQTT_TEST_CLIENT_ENDPOINT`  
GEOFENCE_ARN = `YOUR_GEOFENCE_COLLECTION_NAME`
```

Utilizzando l'app di esempio

Dopo aver impostato il codice di esempio, ora puoi eseguire l'app su un simulatore iOS o su un dispositivo fisico.

1. Crea ed esegui l'app.
2. L'app ti chiederà le autorizzazioni di localizzazione e notifica. Devi consentirli.
3. Premi il pulsante Configurazione Cognito.
4. Salvare la configurazione.
5. Ora puoi visualizzare le opzioni di filtro per tempo, distanza e precisione. Usali secondo le tue necessità.
6. Vai alla scheda Tracciamento dell'app e vedrai la mappa e il pulsante Avvia tracciamento.
7. Se hai installato l'app su un simulatore, potresti voler simulare i cambiamenti di posizione. Questo può essere fatto in Funzionalità nell'opzione del menu Posizione. Ad esempio, seleziona Caratteristiche, poi Posizione, quindi Freeway Drive.
8. Premi il pulsante Avvia tracciamento. Dovresti vedere i punti di tracciamento sulla mappa.
9. L'app monitora anche le posizioni in background. Pertanto, quando sposti l'app in background, ti chiederà l'autorizzazione per continuare il tracciamento in modalità background.
10. Puoi interrompere il tracciamento toccando il pulsante Stop Tracking.

## Taggare le risorse di Amazon Location Service

Usa l'etichettatura delle risorse in Amazon Location per creare tag per classificare le tue risorse in base a scopo, proprietario, ambiente o criteri. L'etichettatura delle risorse ti aiuta a gestire, identificare, organizzare, cercare e filtrare le tue risorse.

Ad esempio, con AWS Resource Groups, puoi creare gruppi di AWS risorse basati su uno o più tag o porzioni di tag. È inoltre possibile creare gruppi in base alla loro occorrenza in uno stack AWS

CloudFormation. Utilizzando Resource Groups e Tag Editor, è possibile consolidare e visualizzare i dati per applicazioni che consistono in più servizi, risorse e regioni in un'unica posizione. Per ulteriori informazioni sulle [strategie di etichettatura comuni](#), consulta l'AWS General Reference.

Ogni tag è un'etichetta composta da una chiave e da un valore definiti dall'utente:

- Chiave tag: un'etichetta generale che classifica i valori dei tag. Ad esempio, `CostCenter`.
- Valore del tag: una descrizione opzionale per la categoria chiave del tag. Ad esempio, `MobileAssetTrackingResourcesProd`.

Questo argomento ti aiuta a iniziare a utilizzare i tag esaminando le restrizioni relative ai tag. Inoltre, mostra come creare tag e utilizzarli per tenere traccia dei costi AWS per ogni tag attivo utilizzando i report di allocazione dei costi.

## Argomenti

- [Restrizioni di tagging](#)
- [Concedi l'autorizzazione per taggare le risorse](#)
- [Aggiungere un tag a una risorsa Amazon Location Service](#)
- [Tieni traccia del costo delle risorse per tag](#)
- [Controlla l'accesso alle risorse di Amazon Location Service utilizzando i tag](#)
- [Ulteriori informazioni](#)

## Restrizioni di tagging

Ai tag si applicano le seguenti limitazioni di base:

- Numero massimo di tag per risorsa: 50
- Per ciascuna risorsa, ogni chiave del tag deve essere univoca e ogni chiave del tag può avere un solo valore.

### Note

Se si aggiunge un tag la cui chiave è la stessa di un tag esistente, il nuovo tag sovrascrive il tag esistente.

- Lunghezza massima della chiave: 128 caratteri Unicode in formato UTF-8

- Lunghezza massima del valore: 256 caratteri Unicode in formato UTF-8
- I caratteri consentiti nei servizi sono: lettere, numeri e spazi rappresentabili in formato UTF-8 e i seguenti caratteri speciali + - = . \_ : / @.
- Per chiavi e valori di tag viene fatta la distinzione tra maiuscole e minuscole.
- Il prefisso `aws:` è riservato per l'uso di AWS. Se il tag ha una chiave di tag con questo prefisso, non puoi modificare o eliminare la chiave o il valore de tag. I tag con il `aws:` prefisso non vengono conteggiati ai fini del limite di tag per risorsa.

## Concedi l'autorizzazione per taggare le risorse

Puoi utilizzare le policy IAM per controllare l'accesso alle tue risorse Amazon Location e concedere l'autorizzazione a taggare una risorsa al momento della creazione. Oltre a concedere l'autorizzazione a creare risorse, la policy può includere `Action` autorizzazioni per consentire operazioni di tagging:

- `geo:TagResource`— Consente a un utente di assegnare uno o più tag a una risorsa Amazon Location specificata.
- `geo:UntagResource`— Consente a un utente di rimuovere uno o più tag da una risorsa Amazon Location specificata.
- `geo:ListTagsForResource`— Consente a un utente di elencare tutti i tag assegnati a una risorsa Amazon Location.

Di seguito è riportato un esempio di policy per consentire a un utente di creare una raccolta di geofence e taggare le risorse:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowTaggingForGeofenceCollectionOnCreation",
      "Effect": "Allow",
      "Action": [
        "geo:CreateGeofenceCollection",
        "geo:TagResource"
      ],
      "Resource": "arn:aws:geo:region:accountID:geofence-collection/*"
    }
  ]
}
```

}

## Aggiungere un tag a una risorsa Amazon Location Service

Puoi aggiungere tag quando crei le tue risorse utilizzando la console Amazon Location, AWS CLI, o le API di Amazon Location:

- [Crea una risorsa cartografica](#)
- [Crea una risorsa per l'indice dei luoghi](#)
- [Crea una risorsa per il calcolo del percorso](#)
- [Crea una collezione di geofence](#)
- [Crea una risorsa di tracciamento](#)

Per etichettare le risorse esistenti, modifica o elimina i tag

1. Apri la console Amazon Location all'[indirizzo https://console.aws.amazon.com/location/](https://console.aws.amazon.com/location/).
2. Nel riquadro di navigazione a sinistra, scegli la risorsa che desideri taggare. Ad esempio, Maps.
3. Scegli una risorsa dall'elenco.
4. Scegli Gestisci tag per aggiungere, modificare o eliminare i tag.

## Tieni traccia del costo delle risorse per tag

Puoi utilizzare i tag per l'allocazione dei costi per tenere traccia AWS dei costi in dettaglio. Dopo aver attivato i tag di allocazione dei costi, AWS utilizza i tag di allocazione dei costi per organizzare la fatturazione delle risorse nel rapporto sull'allocazione dei costi. Ciò consente di classificare e tenere traccia dei costi di utilizzo.

È possibile attivare due tipi di tag di allocazione dei costi:

- [AWS-generati](#): questi tag sono generati da AWS. I tag AWS utilizzano il `aws:` prefisso, ad esempio, `aws:createdBy`.
- [Definito dall'utente](#): si tratta di tag personalizzati creati dall'utente. I tag definiti dall'utente utilizzano il `user:` prefisso, ad esempio, `user:CostCenter`.

È necessario attivare ogni tipo di tag singolarmente. Dopo l'attivazione dei tag, puoi [abilitare AWS Cost Explorer](#) o visualizzare il rapporto mensile sull'allocazione dei costi.

## AWS-generated tags

Per attivare i tag generati da AWS

1. Apri la console di gestione fatturazione e costi all'indirizzo <https://console.aws.amazon.com/billing/>.
2. Nel riquadro di navigazione a sinistra, scegli Cost Allocation Tags.
3. Nella scheda AWS-Tag di allocazione dei costi generati, seleziona le chiavi dei tag che desideri attivare.
4. Seleziona Activate (Attiva).

## User-defined tags

Per attivare i tag definiti dall'utente

1. Apri la console di gestione fatturazione e costi all'indirizzo <https://console.aws.amazon.com/billing/>.
2. Nel riquadro di navigazione a sinistra, scegli Tag di allocazione dei costi.
3. Nella scheda Tag di allocazione dei costi definiti dall'utente, seleziona le chiavi dei tag che desideri attivare.
4. Seleziona Activate (Attiva).

Dopo aver attivato i tag, AWS genera un [rapporto mensile sull'allocazione dei costi](#) per l'utilizzo e i costi delle risorse. Questo rapporto sull'allocazione dei costi include tutti i AWS costi per ogni periodo di fatturazione, comprese le risorse contrassegnate e senza tag. Per ulteriori informazioni, consulta la pagina sull'[utilizzo dei tag per l'allocazione dei costi](#) nella Guida per l'utente di AWS Billing and Cost Management.

## Controlla l'accesso alle risorse di Amazon Location Service utilizzando i tag

AWS Identity and Access ManagementLe policy (IAM) supportano condizioni basate su tag, che consentono di gestire l'autorizzazione per le risorse in base a tag, chiavi e valori specifici. Ad esempio, una policy di ruolo IAM può includere condizioni per limitare l'accesso ad ambienti specifici, come lo sviluppo, il test o la produzione, in base ai tag.

Per ulteriori informazioni, consulta l'argomento sul [controllo dell'accesso alle risorse in base ai tag](#).



## Ulteriori informazioni

Per ulteriori informazioni su:

- Best practice per l'etichettatura, consulta [Tagging delle risorse AWS](#) nell'AWS General Reference.
- Utilizzo dei tag per controllare l'accesso alle AWS risorse, consulta [Controllare l'accesso alle AWS risorse utilizzando i tag nella Guida](#) per l'AWS Identity and Access Management.

## Concessione dell'accesso ad Amazon Location Service

Per utilizzare Amazon Location Service, a un utente deve essere concesso l'accesso alle risorse e alle API che compongono Amazon Location. Esistono tre strategie che puoi utilizzare per concedere l'accesso alle tue risorse.

- Usa IAM: per concedere l'accesso agli utenti autenticati con AWS IAM Identity Center o AWS Identity and Access Management (IAM), crea una policy IAM che consenta l'accesso alle risorse che desideri. Per ulteriori informazioni su IAM e Amazon Location, consulta [Identity and Access Management per Amazon Location Service](#).
- Usa chiavi API: per concedere l'accesso a utenti non autenticati, puoi creare chiavi API che consentono l'accesso in sola lettura alle tue risorse di Amazon Location Service. Ciò è utile nel caso in cui non desideri autenticare tutti gli utenti. Ad esempio, un'applicazione web. Per ulteriori informazioni sulle chiavi API, vedere [Consentire l'accesso degli ospiti non autenticati alla tua applicazione utilizzando le chiavi API](#).
- Usa Amazon Cognito: un'alternativa alle chiavi API consiste nell'utilizzare Amazon Cognito per concedere l'accesso anonimo. Amazon Cognito ti consente di creare un'autorizzazione più completa con policy per definire cosa può essere fatto dagli utenti non autenticati. Per ulteriori informazioni sull'uso di Amazon Cognito, consulta [Consentire l'accesso di ospiti non autenticati alla tua applicazione utilizzando Amazon Cognito](#).

### Note

Puoi anche utilizzare Amazon Cognito per utilizzare il tuo processo di autenticazione o per combinare più metodi di autenticazione, utilizzando Amazon Cognito Federated Identities. Per ulteriori informazioni, consulta [Getting Started with Federated Identities](#) nella Amazon Cognito Developer Guide.

## Argomenti

- [Consentire l'accesso degli ospiti non autenticati alla tua applicazione utilizzando le chiavi API](#)
- [Consentire l'accesso di ospiti non autenticati alla tua applicazione utilizzando Amazon Cognito](#)

## Consentire l'accesso degli ospiti non autenticati alla tua applicazione utilizzando le chiavi API

Quando chiami le API di Amazon Location Service nelle tue applicazioni, in genere effettui questa chiamata come utente autenticato autorizzato a effettuare le chiamate API. Tuttavia, ci sono alcuni casi in cui non desideri autenticare tutti gli utenti della tua applicazione. Ad esempio, potresti volere che un'applicazione web che mostri la sede della tua attività commerciale sia disponibile per tutti gli utenti del sito Web, indipendentemente dal fatto che abbiano effettuato l'accesso o meno. In questo caso, un'alternativa consiste nell'utilizzare le chiavi API per effettuare le chiamate API.

Le chiavi API sono un valore chiave associato a risorse Amazon Location Service specifiche presenti nelle tue Account AWS risorse e ad azioni specifiche che puoi eseguire su tali risorse. Puoi utilizzare una chiave API nella tua applicazione per effettuare chiamate non autenticate alle API di Amazon Location per tali risorse. Ad esempio, se associ una chiave API alla risorsa cartografica MyMap e GetMap\* alle azioni, un'applicazione che utilizza quella chiave API sarà in grado di visualizzare le mappe create con quella risorsa e al tuo account verrà addebitato come qualsiasi altro utilizzo derivante dal tuo account. La stessa chiave API non consentirebbe di modificare o aggiornare la risorsa della mappa: è consentito solo l'utilizzo della risorsa.

### Note

Le chiavi API possono essere utilizzate solo con le risorse di mappe, luoghi e percorsi e non è possibile modificare o creare tali risorse. Se la tua applicazione richiede l'accesso ad altre risorse o azioni per gli utenti non autenticati, puoi utilizzare Amazon Cognito per fornire l'accesso insieme o al posto delle chiavi API. Per ulteriori informazioni, consulta [Consentire l'accesso di ospiti non autenticati alla tua applicazione utilizzando Amazon Cognito](#).

Le chiavi API includono un valore in chiaro che dà accesso a una o più risorse del tuo Account AWS. Se qualcuno copia la tua chiave API, può accedere a quelle stesse risorse. Per evitare ciò, puoi specificare i domini in cui la chiave API può essere utilizzata al momento della creazione della chiave. Questi domini sono chiamati referenti. Se necessario, puoi anche creare chiavi API a breve termine impostando i tempi di scadenza delle tue chiavi API.

## Argomenti

- [Chiavi API a confronto con Amazon Cognito](#)
- [Creazione di chiavi API](#)
- [Utilizzo di una chiave API per chiamare un'API Amazon Location](#)
- [Utilizzo di una chiave API per il rendering di una mappa](#)
- [Gestione della durata delle chiavi API](#)

## Chiavi API a confronto con Amazon Cognito

Le chiavi API e Amazon Cognito vengono utilizzate in modi simili per scenari simili, quindi perché dovresti usarne una rispetto all'altra? L'elenco seguente evidenzia alcune delle differenze tra i due.

- Le chiavi API sono disponibili solo per le risorse relative a mappe, luoghi e percorsi e solo per determinate azioni. Amazon Cognito può essere utilizzato per autenticare l'accesso alla maggior parte delle API di Amazon Location Service.
- Le prestazioni delle richieste di mappe con chiavi API sono in genere più veloci rispetto a scenari simili con Amazon Cognito. Un'autenticazione più semplice significa un minor numero di round trip verso il servizio e di richieste memorizzate nella cache quando si ottiene nuovamente lo stesso riquadro della mappa in un breve periodo di tempo.
- Con Amazon Cognito, puoi utilizzare il tuo processo di autenticazione o combinare più metodi di autenticazione, utilizzando Amazon Cognito Federated Identities. Per ulteriori informazioni, consulta [Getting Started with Federated Identities](#) nella Amazon Cognito Developer Guide.

## Creazione di chiavi API

Puoi creare una chiave API e associarla a una o più risorse del tuo Account AWS.

Puoi creare una chiave API utilizzando la console Amazon Location Service AWS CLI, o le API di Amazon Location.

### Console

Per creare una chiave API utilizzando la console Amazon Location Service

1. Nella [console Amazon Location](#), scegli le chiavi API dal menu a sinistra.
2. Nella pagina delle chiavi API, scegli Crea chiave API.

3. Nella pagina Crea chiave API, inserisci le seguenti informazioni:
  - Nome: un nome per la tua chiave API, ad esempio `MyWebAppKey`.
  - Descrizione: una descrizione opzionale per la tua chiave API.
  - Risorse: scegli le risorse di Amazon Location a cui consentire l'accesso con questa chiave API dal menu a discesa. Puoi aggiungere più di una risorsa selezionando `Aggiungi risorsa`.
  - Azioni: specifica le azioni che desideri autorizzare con questa chiave API. È necessario selezionare almeno un'azione che corrisponda a ciascun tipo di risorsa selezionato. Ad esempio, se hai selezionato una risorsa relativa ai luoghi, devi selezionare almeno una delle scelte in Azioni relative ai luoghi.
  - Ora di scadenza: facoltativamente, aggiungi una data e un'ora di scadenza per la tua chiave API. Per ulteriori informazioni, consulta [Gestione della durata delle chiavi API](#).
  - Referenti: facoltativamente, aggiungi uno o più domini in cui utilizzare la chiave API. Ad esempio, se la chiave API serve a consentire l'esecuzione di un'applicazione sul sito `Webexample.com`, puoi inserirla `*.example.com/` come referente consentito.
  - Tag: facoltativamente, aggiungi tag alla chiave API.
4. Scegli `Crea chiave API` per creare la chiave API.
5. Nella pagina dei dettagli della chiave API, puoi visualizzare le informazioni sulla chiave API che hai creato. Scegli `Mostra chiave API` per vedere il valore della chiave che usi quando chiami le API di Amazon Location. Il valore della chiave avrà il formato `v1.public.a1b2c3d4...`. Per ulteriori informazioni sull'utilizzo della chiave API per il rendering delle mappe, consulta [Utilizzo di una chiave API per il rendering di una mappa](#).

## API

Per creare una chiave API utilizzando le API di Amazon Location

Usa l'[CreateKey](#) operazione dalle API di Amazon Location.

L'esempio seguente è una richiesta API per creare una chiave API chiamata `ExampleKey` senza data di scadenza e accedere a una singola risorsa della mappa.

```
POST /metadata/v0/keys HTTP/1.1
Content-type: application/json

{
  "KeyName": "ExampleKey"
```

```

"Restrictions": {
  "AllowActions": [
    "geo:GetMap*"
  ],
  "AllowResources": [
    "arn:aws:geo:region:map/mapname"
  ]
},
"NoExpiry": true
}
}

```

La risposta include il valore della chiave API da utilizzare per accedere alle risorse delle applicazioni. Il valore chiave avrà il formato `v1.public.a1b2c3d4...`. Per ulteriori informazioni sull'utilizzo della chiave API per il rendering delle mappe, consulta [Utilizzo di una chiave API per il rendering di una mappa](#).

Puoi anche utilizzare l'[DescribeKey](#) API per trovare il valore chiave di una chiave in un secondo momento.

## AWS CLI

Per creare una chiave API utilizzando AWS CLI i comandi

Utilizza il comando [create-key](#).

L'esempio seguente crea una chiave API chiamata *ExampleKey* senza data di scadenza e l'accesso a una singola risorsa della mappa.

```

aws location \
  create-key \
  --key-name ExampleKey \
  --restrictions '{"AllowActions":["geo:GetMap*"],"AllowResources":
["arn:aws:geo:region:map/mapname"]}' \
  --no-expiry

```

La risposta include il valore della chiave API da utilizzare per l'accesso alle risorse nelle applicazioni. Il valore chiave avrà il formato `v1.public.a1b2c3d4...`. Per ulteriori informazioni sull'utilizzo della chiave API per il rendering delle mappe, consulta [Utilizzo di una chiave API per il rendering di una mappa](#). La risposta a `create-key` è simile alla seguente.

```
{
```

```
"Key": "v1.public.a1b2c3d4...",
"KeyArn": "arn:aws:geo:region:accountId:api-key/ExampleKey",
"KeyName": "ExampleKey",
"CreateTime": "2023-02-06T22:33:15.693Z"
}
```

Puoi anche usarlo `describe-key` per trovare il valore chiave in un secondo momento. L'esempio seguente mostra come `describe-key` richiamare una chiave API denominata *ExampleKey*.

```
aws location describe-key \
  --key-name ExampleKey
```

## Utilizzo di una chiave API per chiamare un'API Amazon Location

Dopo aver creato una chiave API, puoi utilizzare il valore della chiave per effettuare chiamate alle API di Amazon Location nella tua applicazione.

Le API che supportano le chiavi API dispongono di un parametro aggiuntivo che assume il valore della chiave API. Ad esempio, se chiami l'`GetPlaceAPI`, puoi inserire il parametro [chiave](#) nel modo seguente

```
GET /places/v0/indexes/IndexName/places/PlaceId?key=KeyValue
```

Se inserisci questo valore, non è necessario autenticare la chiamata API con AWS Sig v4 come faresti normalmente.

Per JavaScript gli sviluppatori, puoi utilizzare Amazon Location [JavaScript Aiutante di autenticazione](#) per facilitare l'autenticazione delle operazioni API con le chiavi API.

Per gli sviluppatori di dispositivi mobili, puoi utilizzare i seguenti SDK di autenticazione mobile Amazon Location:

- [SDK di autenticazione mobile Amazon Location Service per iOS](#)
- [SDK di autenticazione mobile Amazon Location Service per Android](#)

Per AWS CLI gli utenti, quando si utilizza il `--key` parametro, è necessario utilizzare anche il `--no-sign-request` parametro, per evitare di firmare con Sig v4.

**Note**

Se includi sia una firma key and che una AWS Sig v4 in una chiamata ad Amazon Location Service, viene utilizzata solo la chiave API.

## Utilizzo di una chiave API per il rendering di una mappa

È possibile utilizzare il valore della chiave API per eseguire il rendering di una mappa nell'applicazione utilizzando MapLibre. Questo è leggermente diverso rispetto all'utilizzo delle chiavi API in altre API di Amazon Location che chiami direttamente, perché MapLibre effettua quelle chiamate per te.

Il codice di esempio seguente mostra l'utilizzo della chiave API per eseguire il rendering di una mappa in una semplice pagina Web utilizzando il controllo della mappa MapLibre GL JS. *Affinché questo codice funzioni correttamente, sostituisci `v1.public.your-api-key-value`, `us-east-1` e `ExampleMap` con valori corrispondenti ai tuoi. Account AWS*

```
<!-- index.html -->
<html>
  <head>
    <link href="https://unpkg.com/maplibre-gl@1.14.0/dist/maplibre-gl.css"
rel="stylesheet" />
    <style>
      body { margin: 0; }
      #map { height: 100vh; }
    </style>
  </head>
  <body>
    <!-- Map container -->
    <div id="map" />
    <!-- JavaScript dependencies -->
    <script src="https://unpkg.com/maplibre-gl@1.14.0/dist/maplibre-gl.js"></script>
    <script>
      const apiKey = "v1.public.your-api-key-value"; // API key
      const region = "us-east-1"; // Region
      const mapName = "ExampleMap"; // Map name
      // URL for style descriptor
      const styleUrl = `https://maps.geo.${region}.amazonaws.com/maps/v0/maps/
${mapName}/style-descriptor?key=${apiKey}`;
      // Initialize the map
```

```
const map = new maplibregl.Map({
  container: "map",
  style: styleUrl,
  center: [-123.1187, 49.2819],
  zoom: 11,
});
map.addControl(new maplibregl.NavigationControl(), "top-left");
</script>
</body>
</html>
```

## Gestione della durata delle chiavi API

Puoi creare chiavi API che funzionano a tempo indeterminato. Tuttavia, se desideri creare una chiave API temporanea, ruotare le chiavi API regolarmente o revocare una chiave API esistente, puoi utilizzare la scadenza della chiave API.

Quando crei una nuova chiave API o ne aggiorni una esistente, puoi impostare l'ora di scadenza di quella chiave API.

- Quando una chiave API raggiunge la data di scadenza, la chiave viene disattivata automaticamente. Le chiavi inattive non possono più essere utilizzate per effettuare richieste di mappe.
- Puoi eliminare una chiave API 90 giorni dopo averla disattivata.
- Se hai una chiave inattiva che non hai ancora eliminato, puoi ripristinarla aggiornando l'ora di scadenza a un'ora futura.
- Per creare una chiave permanente, puoi rimuovere la data di scadenza.
- Se tenti di disattivare una chiave API che è stata utilizzata negli ultimi 7 giorni, ti verrà richiesto di confermare che desideri apportare la modifica. Se utilizzi l'API Amazon Location Service, o la AWS CLI, riceverai un errore, a meno che non imposti il `ForceUpdate` parametro su `true`.

## Consentire l'accesso di ospiti non autenticati alla tua applicazione utilizzando Amazon Cognito

Puoi utilizzare l'autenticazione Amazon Cognito come alternativa all'utilizzo diretto AWS Identity and Access Management (IAM) sia con SDK frontend che con richieste HTTPS dirette.

Potresti voler utilizzare questa forma di autenticazione per i seguenti motivi:



- **Utenti non autenticati:** se disponi di un sito Web con utenti anonimi, puoi utilizzare i pool di identità di Amazon Cognito. Per ulteriori informazioni, consulta la sezione su. [the section called “Utilizzo di Amazon Cognito”](#)
- **Autenticazione personalizzata:** se desideri utilizzare il tuo processo di autenticazione o combinare più metodi di autenticazione, puoi utilizzare Amazon Cognito Federated Identities. Per ulteriori informazioni, consulta [Getting Started with Federated Identities](#) nella Amazon Cognito Developer Guide.

Amazon Cognito fornisce autenticazione, autorizzazione e gestione degli utenti per app Web e mobili. Puoi utilizzare i pool di identità non autenticati di Amazon Cognito con Amazon Location per consentire alle applicazioni di recuperare credenziali temporanee e limitate. AWS

Per ulteriori informazioni, consulta [Getting Started with User Pools](#) nella Amazon Cognito Developer Guide.

#### Note

Per gli sviluppatori di dispositivi mobili, Amazon Location fornisce SDK di autenticazione mobile per iOS e Android, consulta i seguenti repository github per ulteriori informazioni:

- [SDK di autenticazione mobile Amazon Location Service per iOS](#)
- [SDK di autenticazione mobile Amazon Location Service per Android](#)

## Creazione di un pool di identità in Amazon Cognito

Puoi creare pool di identità Amazon Cognito per consentire l'accesso degli ospiti non autenticati alla tua applicazione tramite la console Amazon Cognito, o AWS CLI e API di Amazon Cognito.

#### Important

Il pool che crei deve trovarsi nella stessa Account AWS AWS regione delle risorse di Amazon Location Service che stai utilizzando.

Puoi utilizzare le policy IAM associate a ruoli di identità non autenticati con le seguenti azioni:

- `geo:GetMap*`

- `geo:SearchPlaceIndex*`
- `geo:GetPlace`
- `geo:CalculateRoute*`
- `geo:GetGeofence`
- `geo:ListGeofences`
- `geo:PutGeofence`
- `geo:BatchDeleteGeofence`
- `geo:BatchPutGeofence`
- `geo:BatchEvaluateGeofences`
- `geo:GetDevicePosition*`
- `geo:ListDevicePositions`
- `geo:BatchDeleteDevicePositionHistory`
- `geo:BatchGetDevicePosition`
- `geo:BatchUpdateDevicePosition`

L'inclusione di altre azioni di Amazon Location non avrà alcun effetto e le identità non autenticate non saranno in grado di chiamarle.

## Example

Per creare un pool di identità utilizzando la console Amazon Cognito

1. Passa alla [console Amazon Cognito](#).
2. Scegli Manage Identity Pools (Gestisci pool di identità).
3. Scegli Crea nuovo pool di identità, quindi inserisci un nome per il tuo pool di identità.
4. Dalla sezione pieghevole Identità non autenticate, scegli Abilita l'accesso alle identità non autenticate.
5. Seleziona Create Pool (Crea pool).
6. Scegli quali ruoli IAM desideri utilizzare con il tuo pool di identità.
7. Espandi Visualizza dettagli.
8. In Identità non autenticate, inserisci il nome del ruolo.
9. Espandi la sezione Visualizza documento di policy, quindi scegli Modifica per aggiungere la tua policy.

## 10. Aggiungi la tua politica per concedere l'accesso alle tue risorse.

Di seguito sono riportati alcuni esempi di policy per Maps, Places, Tracker e Routes. Per utilizzare gli esempi per la tua politica, sostituisci i segnaposto *region* e *AccountID*:

### Maps policy example

La seguente politica concede l'accesso in sola lettura a una risorsa cartografica denominata. *ExampleMap*

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "MapsReadOnly",
      "Effect": "Allow",
      "Action": [
        "geo:GetMapStyleDescriptor",
        "geo:GetMapGlyphs",
        "geo:GetMapSprites",
        "geo:GetMapTile"
      ],
      "Resource": "arn:aws:geo:region:accountID:map/ExampleMap"
    }
  ]
}
```

L'aggiunta di una [condizione IAM](#) corrispondente `aws:referrer` consente di limitare l'accesso del browser alle risorse a un elenco di URL o prefissi URL. L'esempio seguente consente l'accesso a una risorsa cartografica denominata solo `RasterEsriImagery` dal sito `Web: example.com`

#### Warning

Sebbene `aws:referrer` possa limitare l'accesso, non è un meccanismo di sicurezza. È pericoloso includere un valore dell'intestazione del referrer pubblicamente noto. Parti non autorizzate possono utilizzare browser modificati o personalizzati per fornire qualsiasi valore `aws:referrer` scelto. Di conseguenza, non `aws:referrer` deve essere utilizzato per impedire a parti non autorizzate di effettuare richieste dirette. AWS È disponibile solo per consentire ai clienti di proteggere i propri contenuti

digitali, come i contenuti memorizzati su Amazon S3, da riferimenti su siti di terze parti non autorizzate. [Per ulteriori informazioni, vedere: referer.AWS](#)

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "geo:GetMap*",
      "Resource": "arn:aws:geo:us-west-2:111122223333:map/RasterEsriImagery",
      "Condition": {
        "StringLike": {
          "aws:referer": [
            "https://example.com/*",
            "https://www.example.com/*"
          ]
        }
      }
    }
  ]
}
```

Se usi [Tangram per visualizzare una mappa, non utilizza](#) i descrittori di stile, i glifi o gli sprite restituiti dall'API Maps. Viene invece configurato puntando a un file.zip che contiene le regole di stile e le risorse necessarie. La seguente politica concede l'accesso in sola lettura a una risorsa cartografica denominata per l'operazione. *ExampleMapGetMapTile*

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "MapsReadOnly",
      "Effect": "Allow",
      "Action": [
        "geo:GetMapTile"
      ],
      "Resource": "arn:aws:geo:region:accountID:map/ExampleMap"
    }
  ]
}
```

```
}
```

## Places policy example

La seguente politica consente l'accesso in sola lettura a una risorsa dell'indice dei luoghi denominata *ExamplePlaceIndex* per la ricerca di luoghi per testo o posizioni.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "PlacesReadOnly",
      "Effect": "Allow",
      "Action": [
        "geo:SearchPlaceIndex*",
        "geo:GetPlace"
      ],
      "Resource": "arn:aws:geo:region:accountID:place-index/ExamplePlaceIndex"
    }
  ]
}
```

L'aggiunta di una [condizione IAM](#) corrispondente `aws:referrer` consente di limitare l'accesso del browser alle risorse a un elenco di URL o prefissi URL. L'esempio seguente nega l'accesso a una risorsa dell'indice dei luoghi denominata *ExamplePlaceIndex* da tutti i siti Web di riferimento, tranne `example.com`

### Warning

Sebbene `aws:referrer` possa limitare l'accesso, non è un meccanismo di sicurezza. È pericoloso includere un valore dell'intestazione del referrer pubblicamente noto. Parti non autorizzate possono utilizzare browser modificati o personalizzati per fornire qualsiasi valore `aws:referrer` scelto. Di conseguenza, non `aws:referrer` deve essere utilizzato per impedire a parti non autorizzate di effettuare richieste dirette. AWS È disponibile solo per consentire ai clienti di proteggere i propri contenuti digitali, come i contenuti memorizzati su Amazon S3, da riferimenti su siti di terze parti non autorizzate. [Per ulteriori informazioni, vedere: referer.AWS](#)


```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "geo:*",
      "Resource": "arn:aws:geo:us-west-2:111122223333:place-
index/ExamplePlaceIndex",
      "Condition": {
        "StringLike": {
          "aws:referrer": [
            "https://example.com/*",
            "https://www.example.com/*"
          ]
        }
      }
    }
  ]
}
```

### Trackers policy example

La seguente politica consente l'accesso a una risorsa tracker denominata per aggiornare le posizioni dei dispositivi. *ExampleTracker*

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "UpdateDevicePosition",
      "Effect": "Allow",
      "Action": [
        "geo:BatchUpdateDevicePosition"
      ],
      "Resource": "arn:aws:geo:region:accountID:tracker/ExampleTracker"
    }
  ]
}
```

L'aggiunta di una [condizione IAM](#) corrispondente `aws:referrer` consente di limitare l'accesso del browser alle risorse a un elenco di URL o prefissi URL. L'esempio seguente nega l'accesso a una risorsa tracker denominata *ExampleTracker* da tutti i siti Web di riferimento, tranne. `example.com`

 Warning

Sebbene `aws:referrer` possa limitare l'accesso, non è un meccanismo di sicurezza. È pericoloso includere un valore dell'intestazione del referrer pubblicamente noto. Parti non autorizzate possono utilizzare browser modificati o personalizzati per fornire qualsiasi valore `aws:referrer` scelto. Di conseguenza, non `aws:referrer` deve essere utilizzato per impedire a parti non autorizzate di effettuare richieste dirette. AWS È disponibile solo per consentire ai clienti di proteggere i propri contenuti digitali, come i contenuti memorizzati su Amazon S3, da riferimenti su siti di terze parti non autorizzate. [Per ulteriori informazioni, vedere: referer.AWS](#)

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "geo:GetDevice*",
      "Resource": "arn:aws:geo:us-west-2:111122223333:tracker/ExampleTracker",
      "Condition": {
        "StringLike": {
          "aws:referrer": [
            "https://example.com/*",
            "https://www.example.com/*"
          ]
        }
      }
    }
  ]
}
```

## Routes policy example

La seguente politica consente l'accesso a una risorsa di calcolo del percorso denominata per calcolare un percorso. *ExampleCalculator*

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "RoutesReadOnly",
      "Effect": "Allow",
      "Action": [
        "geo:CalculateRoute"
      ],
      "Resource": "arn:aws:geo:region:accountID:route-
calculator/ExampleCalculator"
    }
  ]
}
```

L'aggiunta di una [condizione IAM](#) corrispondente `aws:referrer` consente di limitare l'accesso del browser alle risorse a un elenco di URL o prefissi URL. L'esempio seguente nega l'accesso a un calcolatore di percorso denominato *ExampleCalculator* da tutti i siti Web di riferimento, tranne. `example.com`

### Warning

Sebbene `aws:referrer` possa limitare l'accesso, non è un meccanismo di sicurezza. È pericoloso includere un valore dell'intestazione del referrer pubblicamente noto. Parti non autorizzate possono utilizzare browser modificati o personalizzati per fornire qualsiasi valore `aws:referrer` scelto. Di conseguenza, non `aws:referrer` deve essere utilizzato per impedire a parti non autorizzate di effettuare richieste dirette. AWS È disponibile solo per consentire ai clienti di proteggere i propri contenuti digitali, come i contenuti memorizzati su Amazon S3, da riferimenti su siti di terze parti non autorizzate. [Per ulteriori informazioni, vedere: referer.AWS](#)

```
{
  "Version": "2012-10-17",
```



```

"Statement": [
  {
    "Effect": "Allow",
    "Action": "geo:*",
    "Resource": "arn:aws:geo:us-west-2:111122223333:route-
calculator/ExampleCalculator",
    "Condition": {
      "StringLike": {
        "aws:referrer": [
          "https://example.com/*",
          "https://www.example.com/*"
        ]
      }
    }
  }
]
}

```

#### Note

Sebbene i pool di identità non autenticati siano destinati all'esposizione su siti Internet non protetti, tieni presente che verranno scambiati con credenziali standard a tempo limitato. AWS

È importante definire in modo appropriato i ruoli IAM associati ai pool di identità non autenticati.

11. Scegli Consenti per creare i tuoi pool di identità.

Il pool di identità risultante segue la sintassi `<region>:<GUID>`.

Per esempio:

```
us-east-1:1sample4-5678-90ef-aaaa-1234abcd56ef
```

Per altri esempi di policy specifici per Amazon Location, consulta [the section called “Esempi di policy basate su identità”](#).

## Utilizzo dei pool di identità di Amazon Cognito in JavaScript

L'esempio seguente scambia il pool di identità non autenticato creato con credenziali che vengono poi utilizzate per recuperare il descrittore di stile per la risorsa della mappa. *ExampleMap*

```
const AWS = require("aws-sdk");

const credentials = new AWS.CognitoIdentityCredentials({
  IdentityPoolId: "<identity pool ID>" // for example, us-east-1:1sample4-5678-90ef-
  aaaa-1234abcd56ef
});

const client = new AWS.Location({
  credentials,
  region: AWS.config.region || "<region>"
});

console.log(await client.getMapStyleDescriptor("ExampleMap").promise());
```

### Note

Le credenziali recuperate da identità non autenticate sono valide per un'ora.

Di seguito è riportato un esempio di funzione che rinnova automaticamente le credenziali prima della scadenza.

```
async function refreshCredentials() {
  await credentials.refreshPromise();
  // schedule the next credential refresh when they're about to expire
  setTimeout(refreshCredentials, credentials.expireTime - new Date());
}
```

Per semplificare questo lavoro, puoi utilizzare Amazon Location [JavaScript Aiutante di autenticazione](#). Questo serve sia per ottenere le credenziali che per aggiornarle. Questo esempio utilizza l' AWS SDK per la versione 3. JavaScript

```
import { LocationClient, GetMapStyleDescriptorCommand } from "@aws-sdk/client-
location";
import { withIdentityPoolId } from "@aws/amazon-location-utilities-auth-helper";
```

```
const identityPoolId = "<identity pool ID>"; // for example, us-
east-1:1sample4-5678-90ef-aaaa-1234abcd56ef

// Create an authentication helper instance using credentials from Cognito
const authHelper = await withIdentityPoolId(identityPoolId);

const client = new LocationClient({
  region: "<region>", // The region containing both the identity pool and tracker
  resource
  ...authHelper.getLocationClientConfig(), // Provides configuration required to make
  requests to Amazon Location
});

const input = {
  MapName: "ExampleMap",
};

const command = new GetMapStyleDescriptorCommand(input);

console.log(await client.send(command));
```

## Passaggi successivi

- Per modificare i tuoi ruoli, accedi alla console [IAM](#).
- Per gestire i tuoi pool di identità, accedi alla console [Amazon Cognito](#).

## Monitoraggio di Amazon Location Service

Quando usi Amazon Location Service, puoi monitorare l'utilizzo e le risorse nel tempo utilizzando:

- Amazon CloudWatch. Monitora le risorse di Amazon Location Service e fornisce metriche con statistiche quasi in tempo reale.
- AWS CloudTrail. Fornisce il monitoraggio degli eventi di tutte le chiamate alle API di Amazon Location Service.

Questa sezione fornisce informazioni sull'uso di questi servizi.

### Argomenti

- [Monitoraggio di Amazon Location Service con Amazon CloudWatch](#)

- [Registrazione e monitoraggio con AWS CloudTrail](#)

## Monitoraggio di Amazon Location Service con Amazon CloudWatch

Amazon CloudWatch monitora AWS le tue risorse e le applicazioni su cui esegui quasi AWS in tempo reale. Puoi monitorare le risorse di Amazon Location utilizzando CloudWatch, che raccoglie dati grezzi ed elabora i parametri in statistiche significative quasi in tempo reale. Puoi visualizzare informazioni storiche per un massimo di 15 mesi o cercare metriche da visualizzare nella CloudWatch console Amazon per avere una prospettiva più approfondita sulle tue risorse di Amazon Location. Puoi anche impostare allarmi definendo soglie e inviare notifiche o intraprendere azioni quando tali soglie vengono raggiunte.

Per ulteriori informazioni, consulta la [Amazon CloudWatch User Guide](#)

### Argomenti

- [Metriche di Amazon Location Service esportate su Amazon CloudWatch](#)
- [Visualizza i parametri di Amazon Location Service](#)
- [Crea CloudWatch allarmi per le metriche di Amazon Location Service](#)
- [Utilizzato CloudWatch per monitorare l'utilizzo rispetto alle quote](#)
- [CloudWatch esempi di metriche per Amazon Location Service](#)

## Metriche di Amazon Location Service esportate su Amazon CloudWatch

Le metriche sono punti dati ordinati nel tempo che vengono esportati in CloudWatch. Una dimensione è una coppia nome/valore che identifica la metrica. Per ulteriori informazioni, consulta [Using CloudWatch metrics](#) and [CloudWatch dimensions](#) nella Amazon CloudWatch User Guide.

Di seguito sono riportate le metriche in cui Amazon Location Service esporta CloudWatch nel AWS/Location namespace.

Parametro	Descrizione
CallCount	Il numero di chiamate effettuate a un determinato endpoint API.  Dimensioni valide: nomi delle API di Amazon Location Service  Statistiche valide: Sum

Parametro	Descrizione
	Unità: numero
ErrorCount	<p>Il numero di risposte di errore derivanti dalle chiamate effettuate a un determinato endpoint API.</p> <p>Dimensioni valide: nomi delle API di Amazon Location Service</p> <p>Statistiche valide: Sum</p> <p>Unità: numero</p>
SuccessCount	<p>Il numero di chiamate riuscite effettuate a un determinato endpoint API.</p> <p>Dimensioni valide: nomi delle API di Amazon Location Service</p> <p>Statistiche valide: Sum</p> <p>Unità: numero</p>
CallLatency	<p>Il tempo impiegato dall'operazione per elaborare e restituire una risposta quando viene effettuata una chiamata a un determinato endpoint API.</p> <p>Dimensioni valide: nomi delle API di Amazon Location Service</p> <p>Statistiche valide: media</p> <p>Unità: millisecondi</p>

## Visualizza i parametri di Amazon Location Service

Puoi visualizzare le metriche per Amazon Location Service sulla Amazon CloudWatch console o utilizzando l'Amazon CloudWatchAPI.

Per visualizzare le metriche utilizzando la console CloudWatch

### Example

1. Apri la CloudWatch console all'indirizzo <https://console.aws.amazon.com/cloudwatch/>.
2. Nel riquadro di navigazione, seleziona Parametri.

3. Nella scheda Tutte le metriche, scegli lo spazio dei nomi Amazon Location.
4. Seleziona il tipo di metrica da visualizzare.
5. Seleziona la metrica e aggiungila al grafico.

Per ulteriori informazioni, consulta [Visualizza i parametri disponibili](#) nella Amazon CloudWatch User Guide.

## Crea CloudWatch allarmi per le metriche di Amazon Location Service

Puoi utilizzarlo CloudWatch per impostare allarmi in base alle metriche di Amazon Location Service. Ad esempio, puoi creare un allarme per inviare un'e-mail ogni volta che CloudWatch si verifica un picco di conteggio degli errori.

Gli argomenti seguenti forniscono una panoramica su come impostare gli allarmi utilizzando CloudWatch. Per istruzioni dettagliate, consulta [Using Alarms](#) nella Amazon CloudWatch User Guide.

Per impostare allarmi utilizzando la console CloudWatch

### Example

1. Apri la CloudWatch console all'indirizzo <https://console.aws.amazon.com/cloudwatch/>.
2. Nel pannello di navigazione, scegli Allarme.
3. Scegli Crea allarme.
4. Scegli Select Metric (Seleziona parametro).
5. Nella scheda Tutte le metriche, seleziona lo spazio dei nomi Amazon Location.
6. Seleziona una categoria di metriche.
7. Trova la riga con la metrica per cui desideri creare un allarme, quindi seleziona la casella di controllo accanto a questa riga.
8. Scegli Select Metric (Seleziona parametro).
9. In Metrica, inserisci i valori.
10. Specificare le condizioni di allarme.
11. Seleziona Avanti.
12. Se desideri inviare una notifica quando le condizioni di allarme sono soddisfatte:
  - In Attivazione dello stato di allarme, seleziona lo stato di allarme per richiedere l'invio di una notifica.

- In **Seleziona un argomento SNS**, scegli **Crea nuovo argomento** per creare un nuovo argomento Amazon Simple Notification Service (Amazon SNS). Inserisci il nome dell'argomento e l'e-mail a cui inviare la notifica.
- In **Invia una notifica** per inserire altri indirizzi e-mail a cui inviare la notifica.
- Scegliere **Add notification (Aggiungi notifica)**. Questo elenco viene salvato e visualizzato nel campo per allarmi futuri.

13 Al termine, scegli **Avanti**.

14 Inserisci un nome e una descrizione per l'avviso, quindi scegli **Avanti**.

15 Conferma i dettagli dell'allarme, quindi scegli **Avanti**.

#### Note

Quando crei un nuovo argomento Amazon SNS, devi verificare l'indirizzo e-mail prima di poter inviare una notifica. Se l'e-mail non viene verificata, la notifica non verrà ricevuta quando viene avviato un allarme a causa di un cambiamento di stato.

Per ulteriori informazioni su come impostare gli allarmi utilizzando la CloudWatch console, consulta [Create an Alarm that Sends Email](#) nella Amazon CloudWatch User Guide.

## Utilizzato CloudWatch per monitorare l'utilizzo rispetto alle quote

Puoi creare CloudWatch allarmi Amazon per avvisarti quando l'utilizzo di una determinata quota supera una soglia configurabile. In questo modo puoi riconoscere quando sei vicino ai limiti di quota e adattarne l'utilizzo per evitare superamenti dei costi o richiedere un aumento della quota, se necessario. Per informazioni su come utilizzare per CloudWatch monitorare le quote, consulta la sezione [Visualizzazione delle quote di servizio e impostazione degli allarmi](#) nella Amazon User Guide. CloudWatch

## CloudWatch esempi di metriche per Amazon Location Service

Puoi utilizzare l'[GetMetricData](#) API per recuperare i parametri per Amazon Location.

- Ad esempio, puoi monitorare `CallCount` e impostare un allarme quando si verifica un calo del numero.

Il monitoraggio delle CallCount metriche SendDeviceLocation può aiutarti a darti una prospettiva sugli asset tracciati. Se il dato CallCount diminuisce, significa che gli asset tracciati, come una flotta di camion, hanno smesso di inviare la loro posizione attuale. L'impostazione di un allarme in tal senso può aiutarti a notificare che si è verificato un problema.

- Un altro esempio, è possibile monitorare ErrorCount e impostare un allarme quando si verifica un picco di numero.

I tracker devono essere associati alle raccolte di geofence affinché le posizioni dei dispositivi possano essere valutate rispetto ai geofence. Se disponi di un parco dispositivi che richiede aggiornamenti continui sulla posizione, se visualizzi BatchEvaluateGeofence o riduci a zero significa che gli CallCount BatchPutDevicePosition aggiornamenti non sono più disponibili.

Di seguito è riportato un esempio di output per [GetMetricData](#) con le metriche per CallCount e ErrorCount per la creazione di risorse cartografiche.

```
{
  "StartTime": 1518867432,
  "EndTime": 1518868032,
  "MetricDataQueries": [
    {
      "Id": "m1",
      "MetricStat": {
        "Metric": {
          "Namespace": "AWS/Location",
          "MetricName": "CallCount",
          "Dimensions": [
            {
              "Name": "SendDeviceLocation",
              "Value": "100"
            }
          ]
        },
        "Period": 300,
        "Stat": "SampleCount",
        "Unit": "Count"
      }
    },
    {
      "Id": "m2",
      "MetricStat": {
```



```
    "Metric": {
      "Namespace": "AWS/Location",
      "MetricName": "ErrorCount",
      "Dimensions": [
        {
          "Name": "AssociateTrackerConsumer",
          "Value": "0"
        }
      ]
    },
    "Period": 1,
    "Stat": "SampleCount",
    "Unit": "Count"
  }
}
]
```

## Registrazione e monitoraggio con AWS CloudTrail

AWS CloudTrail è un servizio che fornisce un registro delle azioni intraprese da un utente, un ruolo o un AWS servizio. CloudTrail registra tutte le chiamate API come eventi. Puoi utilizzare Amazon Location Service con CloudTrail per monitorare le tue chiamate API, che includono le chiamate dalla console di Amazon Location Service e le chiamate AWS SDK alle operazioni dell'API Amazon Location Service.

Quando crei un trail, puoi abilitare la distribuzione continua di CloudTrail eventi a un bucket S3, inclusi gli eventi per Amazon Location Service. Se non configuri un percorso, puoi comunque visualizzare gli eventi più recenti nella CloudTrail console nella cronologia degli eventi. Utilizzando le informazioni raccolte da CloudTrail, puoi determinare la richiesta che è stata effettuata ad Amazon Location Service, l'indirizzo IP da cui è stata effettuata la richiesta, chi ha effettuato la richiesta, quando è stata effettuata e dettagli aggiuntivi.

Per ulteriori informazioni in merito CloudTrail, consulta la [Guida AWS CloudTrail per l'utente](#).

### Argomenti

- [Informazioni su Amazon Location Service in CloudTrail](#)
- [Informazioni sulle voci dei file di registro di Amazon Location Service](#)

## Informazioni su Amazon Location Service in CloudTrail

CloudTrail è abilitato sul tuo AWS account al momento della creazione dell'account. Quando si verifica un'attività in Amazon Location Service, tale attività viene registrata in un CloudTrail evento insieme ad altri eventi AWS di servizio nella cronologia degli eventi. È possibile visualizzare, cercare e scaricare gli eventi recenti nell'account AWS. Per ulteriori informazioni, consulta [Visualizzazione degli eventi con la cronologia degli CloudTrail eventi](#).

Per una registrazione continua degli eventi nel tuo AWS account, inclusi gli eventi per Amazon Location Service, crea un percorso. Un trail consente di CloudTrail inviare file di registro a un bucket S3. Per impostazione predefinita, quando si crea un trail nella console, il trail sarà valido in tutte le regioni AWS. Il trail registra gli eventi di tutte le regioni nella partizione AWS e distribuisce i file di log nel bucket S3 specificato. Inoltre, puoi configurare altri AWS servizi per analizzare ulteriormente e agire in base ai dati sugli eventi raccolti nei CloudTrail log.

Per ulteriori informazioni, consulta gli argomenti seguenti:

- [Panoramica della creazione di un trail](#)
- [CloudTrail Servizi e integrazioni supportati](#)
- [Configurazione delle notifiche Amazon SNS per CloudTrail](#)
- [Ricezione di file di CloudTrail registro da più regioni](#) e [ricezione di file di CloudTrail registro da più account](#)

Tutte le azioni di Amazon Location Service vengono registrate CloudTrail e documentate nei riferimenti [API di Amazon Location Service](#). Ad esempio, le chiamate alle `CreateTracker` `DescribeTracker` azioni `UpdateTracker` e generano voci nei file di CloudTrail registro.

Ogni evento o voce di log contiene informazioni sull'utente che ha generato la richiesta. Le informazioni sull'identità consentono di stabilire se la richiesta è stata effettuata:

- Con credenziali utente root o AWS Identity and Access Management (IAM).
- Con credenziali di sicurezza temporanee per un ruolo o un utente federato.
- Da un altro servizio AWS.

Per ulteriori informazioni, consulta [Elemento CloudTrail userIdentity](#).

## Informazioni sulle voci dei file di registro di Amazon Location Service

Un trail è una configurazione che consente la consegna di eventi come file di registro a un bucket S3 specificato dall'utente o ad Amazon CloudWatch Logs. Per ulteriori informazioni, consulta [Working with CloudTrail log files](#) nella Guida per l'AWS CloudTrailutente.

CloudTrail i file di registro contengono una o più voci di registro. Un evento rappresenta una singola richiesta proveniente da qualsiasi fonte e include informazioni sull'operazione desiderata, la data e l'ora della stessa, i parametri della richiesta e così via.

### Note

CloudTrail i file di registro non sono una traccia ordinata delle chiamate API pubbliche, quindi non vengono visualizzati in un ordine specifico. Per determinare l'ordine delle operazioni, usa [eventTime](#).

L'esempio seguente mostra una voce di CloudTrail registro che dimostra l'CreateTrackeroperazione, che crea una risorsa tracker.

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "123456789012",
    "arn": "arn:aws:geo:us-east-1:123456789012:tracker/ExampleTracker",
    "accountId": "123456789012",
    "accessKeyId": "123456789012",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "123456789012",
        "arn": "arn:aws:geo:us-east-1:123456789012:tracker/ExampleTracker",
        "accountId": "123456789012",
        "userName": "exampleUser",
      },
      "webIdFederationData": {},
      "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2020-10-22T16:36:07Z"
      }
    }
  }
}
```

```

    }
  },
  "eventTime": "2020-10-22T17:43:30Z",
  "eventSource": "geo.amazonaws.com",
  "eventName": "CreateTracker",
  "awsRegion": "us-east-1",
  "sourceIPAddress": "192.0.2.0/24-TEST-NET-1",
  "userAgent": "aws-internal/3 aws-sdk-java/1.11.864
Linux/4.14.193-110.317.amzn2.x86_64 OpenJDK_64-Bit_Server_VM/11.0.8+10-LTS java/11.0.8
kotlin/1.3.72 vendor/Amazon.com_Inc. exec-env/AWS_Lambda_java11",
  "requestParameters": {
    "TrackerName": "ExampleTracker",
    "Description": "Resource description"
  },
  "responseElements": {
    "TrackerName": "ExampleTracker",
    "Description": "Resource description",
    "TrackerArn": "arn:partition:service:region:account-id:resource-id",
    "CreateTime": "2020-10-22T17:43:30.521Z"
  },
  "requestID": "557ec619-0674-429d-8e2c-eba0d3f34413",
  "eventID": "3192bc9c-3d3d-4976-bbef-ac590fa34f2c",
  "readOnly": false,
  "eventType": "AwsApiCall",
  "recipientAccountId": "123456789012",
}

```

Di seguito viene mostrata una voce di registro relativa all'DescribeTracker operazione, che restituisce i dettagli di una risorsa tracker.

```

{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "123456789012",
    "arn": "arn:partition:service:region:account-id:resource-id",
    "accountId": "123456789012",
    "accessKeyId": "123456789012",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "123456789012",
        "arn": "arn:partition:service:region:account-id:resource-id",

```

```

        "accountId": "123456789012",
        "userName": "exampleUser",
    },
    "webIdFederationData": {},
    "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2020-10-22T16:36:07Z"
    }
}
},
"eventTime": "2020-10-22T17:43:33Z",
"eventSource": "geo.amazonaws.com",
"eventName": "DescribeTracker",
"awsRegion": "us-east-1",
"sourceIPAddress": "192.0.2.0/24-TEST-NET-1",
"userAgent": "aws-internal/3 aws-sdk-java/1.11.864
Linux/4.14.193-110.317.amzn2.x86_64 OpenJDK_64-Bit_Server_VM/11.0.8+10-LTS java/11.0.8
kotlin/1.3.72 vendor/Amazon.com_Inc. exec-env/AWS_Lambda_java11",
"requestParameters": {
    "TrackerName": "ExampleTracker"
},
"responseElements": null,
"requestID": "997d5f93-cfef-429a-bbed-daab417ceab4",
"eventID": "d9e0eebe-173c-477d-b0c9-d1d8292da103",
"readOnly": true,
"eventType": "AwsApiCall",
"recipientAccountId": "123456789012",
}

```

## Creazione di risorse Amazon Location Service con AWS CloudFormation

Amazon Location Service è integrato con AWS CloudFormation, un servizio che ti aiuta a modellare e configurare AWS le tue risorse in modo da dedicare meno tempo alla creazione e alla gestione delle risorse e dell'infrastruttura. Crei un modello che descrive tutte le AWS risorse che desideri (come le risorse di Amazon Location) e fornisce AWS CloudFormation e configura tali risorse per te.

Quando lo utilizzi AWS CloudFormation, puoi riutilizzare il modello per configurare le risorse di Amazon Location in modo coerente e ripetuto. Descrivere le risorse una volta e quindi allestisci le stesse risorse più volte in più regioni e account AWS.

## Ubicazione e AWS CloudFormation modelli Amazon

Per fornire e configurare le risorse per Amazon Location e i servizi correlati, devi conoscere [AWS CloudFormationi modelli](#). I modelli sono file di testo formattati in JSON o YAML. Questi modelli descrivono le risorse di cui intendi effettuare il provisioning negli stack AWS CloudFormation. Se non hai familiarità con JSON o YAML, puoi usare AWS CloudFormationDesigner per iniziare a utilizzare i modelli AWS CloudFormation. Per ulteriori informazioni, consulta [Che cos'è AWS CloudFormationDesigner?](#) nella Guida per l'utente di AWS CloudFormation.

Amazon Location supporta la creazione dei seguenti tipi di risorse inAWS CloudFormation:

- [AWS::Location::Map](#)
- [AWS::Location::PlaceIndex](#)
- [AWS::Location::RouteCalculator](#)
- [AWS::Location::Tracker](#)
- [AWS::Location::TrackerConsumer](#)
- [AWS::Location::GeofenceCollection](#)

Per ulteriori informazioni, inclusi esempi di modelli JSON e YAML per le risorse di Amazon Location, consulta il [riferimento al tipo di risorsa Amazon Location Service](#) nella Guida per l'AWS CloudFormationutente.

## Ulteriori informazioni su AWS CloudFormation

Per ulteriori informazioni su AWS CloudFormation, consulta le seguenti risorse:

- [AWS CloudFormation](#)
- [Guida per l'utente di AWS CloudFormation](#)
- [Documentazione di riferimento dell'API AWS CloudFormation](#)
- [Guida per l'utente dell'interfaccia a riga di comando di AWS CloudFormation](#)

# Sicurezza in Amazon Location Service

La sicurezza del cloud AWS è la massima priorità. In qualità di AWS cliente, puoi beneficiare di data center e architetture di rete progettati per soddisfare i requisiti delle organizzazioni più sensibili alla sicurezza.

La sicurezza è una responsabilità condivisa tra te e te. AWS Il [modello di responsabilità condivisa](#) descrive questo aspetto come sicurezza del cloud e sicurezza nel cloud:

- **Sicurezza del cloud:** AWS è responsabile della protezione dell'infrastruttura che gestisce AWS i servizi in Cloud AWS. AWS fornisce inoltre servizi che è possibile utilizzare in modo sicuro. I revisori esterni testano e verificano regolarmente l'efficacia della nostra sicurezza nell'ambito dei [AWS Programmi di AWS conformità dei Programmi di conformità](#) dei di . Per maggiori informazioni sui programmi di conformità che si applicano ad Amazon Location Service, consulta [AWS Services in Scope by Compliance Program AWS](#) .
- **Sicurezza nel cloud:** la tua responsabilità è determinata dal AWS servizio che utilizzi. Sei anche responsabile di altri fattori, tra cui la riservatezza dei dati, i requisiti della tua azienda e le leggi e normative vigenti.

Questa documentazione ti aiuta a capire come applicare il modello di responsabilità condivisa quando usi Amazon Location. I seguenti argomenti mostrano come configurare Amazon Location per soddisfare i tuoi obiettivi di sicurezza e conformità. Scopri anche come utilizzare altri AWS servizi che ti aiutano a monitorare e proteggere le tue risorse di Amazon Location.

## Argomenti

- [Protezione dei dati in Amazon Location Service](#)
- [Identity and Access Management per Amazon Location Service](#)
- [Risposta agli incidenti in Amazon Location Service](#)
- [Convalida della conformità per Amazon Location Service](#)
- [Resilienza in Amazon Location Service](#)
- [Sicurezza dell'infrastruttura in Amazon Location Service](#)
- [Configurazione e analisi delle vulnerabilità in Amazon Location](#)
- [Prevenzione del problema "confused deputy" tra servizi](#)
- [Best practice di sicurezza per Amazon Location Service](#)

- [Le migliori pratiche per Amazon Location Service](#)

## Protezione dei dati in Amazon Location Service

Il modello di [responsabilità AWS condivisa modello](#) si applica alla protezione dei dati in Amazon Location Service. Come descritto in questo modello, AWS è responsabile della protezione dell'infrastruttura globale che gestisce tutti i Cloud AWS. L'utente è responsabile del controllo dei contenuti ospitati su questa infrastruttura. L'utente è inoltre responsabile della configurazione della protezione e delle attività di gestione per i Servizi AWS utilizzati. Per ulteriori informazioni sulla privacy dei dati, vedi le [Domande frequenti sulla privacy dei dati](#). Per informazioni sulla protezione dei dati in Europa, consulta il post del blog relativo al [Modello di responsabilità condivisa AWS e GDPR](#) nel Blog sulla sicurezza AWS .

Ai fini della protezione dei dati, consigliamo di proteggere Account AWS le credenziali e configurare i singoli utenti con AWS IAM Identity Center or AWS Identity and Access Management (IAM). In tal modo, a ogni utente verranno assegnate solo le autorizzazioni necessarie per svolgere i suoi compiti. Ti suggeriamo, inoltre, di proteggere i dati nei seguenti modi:

- Utilizza l'autenticazione a più fattori (MFA) con ogni account.
- Usa SSL/TLS per comunicare con le risorse. AWS È richiesto TLS 1.2 ed è consigliato TLS 1.3.
- Configura l'API e la registrazione delle attività degli utenti con. AWS CloudTrail
- Utilizza soluzioni di AWS crittografia, insieme a tutti i controlli di sicurezza predefiniti all'interno Servizi AWS.
- Utilizza i servizi di sicurezza gestiti avanzati, come Amazon Macie, che aiutano a individuare e proteggere i dati sensibili archiviati in Amazon S3.
- Se hai bisogno di moduli crittografici convalidati FIPS 140-2 per l'accesso AWS tramite un'interfaccia a riga di comando o un'API, utilizza un endpoint FIPS. Per ulteriori informazioni sugli endpoint FIPS disponibili, consulta il [Federal Information Processing Standard \(FIPS\) 140-2](#).

Ti consigliamo vivamente di non inserire mai informazioni riservate o sensibili, ad esempio gli indirizzi e-mail dei clienti, nei tag o nei campi di testo in formato libero, ad esempio nel campo Nome. Ciò include quando lavori con Amazon Location o altro Servizi AWS utilizzando la console, l'API o AWS gli SDK. AWS CLI I dati inseriti nei tag o nei campi di testo in formato libero utilizzati per i nomi possono essere utilizzati per i la fatturazione o i log di diagnostica. Quando fornisci un URL a un server esterno, ti suggeriamo vivamente di non includere informazioni sulle credenziali nell'URL per convalidare la tua richiesta al server.



## Privacy dei dati

Con Amazon Location Service, mantieni il controllo dei dati della tua organizzazione. Amazon Location rende anonime tutte le richieste inviate ai fornitori di dati rimuovendo i metadati dei clienti e le informazioni sull'account.

Amazon Location non utilizza fornitori di dati per il tracciamento e il geofencing. Ciò significa che i tuoi dati sensibili rimangono nel tuo AWS account. Questo aiuta a proteggere le informazioni sensibili sulla posizione, come la struttura, le risorse e l'ubicazione del personale, da terze parti, a proteggere la privacy degli utenti e a ridurre il rischio per la sicurezza dell'applicazione.

Per ulteriori informazioni, consulta le [Domande frequenti sulla privacy AWS dei dati](#).

## Conservazione dei dati in Amazon Location

Le seguenti caratteristiche si riferiscono al modo in cui Amazon Location raccoglie e archivia i dati per il servizio:

- Amazon Location Service Trackers: quando utilizzi le API Trackers per tracciare la posizione delle entità, le loro coordinate possono essere memorizzate. Le posizioni dei dispositivi vengono archiviate per 30 giorni prima di essere eliminate dal servizio.
- Amazon Location Service Geofences: quando utilizzi le API Geofences per definire aree di interesse, il servizio memorizza le geometrie fornite. Devono essere eliminati in modo esplicito.

### Note

L'eliminazione AWS dell'account comporta l'eliminazione di tutte le risorse al suo interno. Per ulteriori informazioni, consulta le [Domande frequenti sulla privacy AWS dei dati](#).

## Crittografia dei dati a riposo per Amazon Location Service

Amazon Location Service fornisce la crittografia di default per proteggere i dati sensibili dei clienti archiviati utilizzando chiavi AWS di crittografia proprietarie.

- AWS chiavi di proprietà: Amazon Location utilizza queste chiavi per impostazione predefinita per crittografare automaticamente i dati di identificazione personale. Non puoi visualizzare, gestire o utilizzare chiavi AWS di proprietà o controllarne l'utilizzo. Tuttavia, non è necessario effettuare

alcuna operazione o modificare programmi per proteggere le chiavi che eseguono la crittografia dei dati. Per ulteriori informazioni, consulta la pagina [chiavi di proprietàAWS](#) nella Guida per gli sviluppatori di AWS Key Management Service .

La crittografia predefinita dei dati a riposo aiuta a ridurre il sovraccarico operativo e la complessità associati alla protezione dei dati sensibili. Allo stesso tempo, consente di creare applicazioni sicure che soddisfano i rigorosi requisiti normativi e di conformità alla crittografia.

Sebbene non sia possibile disabilitare questo livello di crittografia o selezionare un tipo di crittografia alternativo, è possibile aggiungere un secondo livello di crittografia rispetto alle chiavi di crittografia esistenti AWS scegliendo una chiave gestita dal cliente quando si creano le risorse di raccolta di tracker e geofence:

- Chiavi gestite dal cliente: Amazon Location supporta l'uso di una chiave simmetrica gestita dal cliente che puoi creare, possedere e gestire per aggiungere un secondo livello di crittografia rispetto alla crittografia di AWS proprietà esistente. Avendo il pieno controllo di questo livello di crittografia, è possibile eseguire operazioni quali:
  - Stabilire e mantenere le policy delle chiavi
  - Stabilire e mantenere le policy e le sovvenzioni IAM
  - Abilitare e disabilitare le policy delle chiavi
  - Ruotare i materiali crittografici delle chiavi
  - Aggiungere tag
  - Creare alias delle chiavi
  - Pianificare l'eliminazione delle chiavi

Per ulteriori informazioni, consulta la [chiave gestita dal cliente nella Guida per](#) gli AWS Key Management Service sviluppatori.

La tabella seguente riassume il modo in cui Amazon Location crittografa i dati di identificazione personale.

Tipo di dati	AWS crittografia a chiave proprietaria	Crittografia con chiavi gestite dal cliente (opzionale)
Position	Abilitato	Abilitato

Tipo di dati	AWS crittografia a chiave proprietaria	Crittografia con chiavi gestite dal cliente (opzionale)
<p>Una geometria puntuale contenente <a href="#">i dettagli sulla posizione del dispositivo</a>.</p>		
<p>PositionProperties</p> <p>Un insieme di coppie chiave-valore <a href="#">associate all'aggiornamento della posizione</a>.</p>	Abilitato	Abilitato
<p>GeofenceGeometry</p> <p>Una geometria di <a href="#">geofence poligonale che rappresenta l'area geofrecintata</a>.</p>	Abilitato	Abilitato
<p>DeviceId</p> <p>L'identificatore del dispositivo specificato durante il <a href="#">caricamento</a> di un aggiornamento della posizione del dispositivo su una risorsa tracker.</p>	Abilitato	Non supportato
<p>GeofenceId</p> <p>Un identificatore specifico quando si <a href="#">memorizza una geometria di geofence</a> o un <a href="#">batch di geofence in una determinata raccolta di geofence</a>.</p>	Abilitato	Non supportato

### Note

Amazon Location abilita automaticamente la crittografia a riposo utilizzando chiavi AWS proprietarie per proteggere gratuitamente i dati di identificazione personale.

Tuttavia, l'utilizzo di una chiave gestita dal cliente comporta dei costi AWS KMS. Per ulteriori informazioni sui prezzi, consulta i [AWS Key Management Service prezzi](#).

Per ulteriori informazioni su AWS KMS, consulta [Cos'è AWS Key Management Service?](#)

## In che modo Amazon Location Service utilizza le sovvenzioni in AWS KMS

Amazon Location richiede una [concessione](#) per utilizzare la tua chiave gestita dal cliente.

Quando crei una [risorsa di tracciamento](#) o una [raccolta di geofence](#) crittografata con una chiave gestita dal cliente, Amazon Location crea una concessione per tuo conto inviando una [CreateGrant](#) richiesta a AWS KMS. Le sovvenzioni AWS KMS vengono utilizzate per consentire ad Amazon Location di accedere a una chiave KMS in un account cliente.

Amazon Location richiede la concessione dell'utilizzo della chiave gestita dal cliente per le seguenti operazioni interne:

- Invia [DescribeKey](#) richieste AWS KMS a per verificare che l'ID della chiave KMS simmetrica gestita dal cliente inserito durante la creazione di un tracker o di una raccolta di geofence sia valido.
- Invia [GenerateDataKeyWithoutPlaintext](#) richieste per generare chiavi dati AWS KMS crittografate dalla chiave gestita dal cliente.
- Invia le richieste [Decrypt](#) a per AWS KMS decrittografare le chiavi di dati crittografate in modo che possano essere utilizzate per crittografare i dati.

Puoi revocare l'accesso alla concessione o rimuovere l'accesso del servizio alla chiave gestita dal cliente in qualsiasi momento. In tal caso, Amazon Location non sarà in grado di accedere a nessuno dei dati crittografati dalla chiave gestita dal cliente, il che influirà sulle operazioni che dipendono da tali dati. Ad esempio, se tenti di [ottenere le posizioni dei dispositivi](#) da un tracker crittografato a cui Amazon Location non può accedere, l'operazione restituirà un `AccessDeniedException` errore.

## Creazione di una chiave gestita dal cliente

Puoi creare una chiave simmetrica gestita dal cliente utilizzando le AWS Management Console, o le API AWS KMS.

## Per creare una chiave simmetrica gestita dal cliente

Segui la procedura riportata in [Creazione di una chiave simmetrica gestita dal cliente](#) nella Guida per gli sviluppatori di AWS Key Management Service .

### Policy della chiave

Le policy della chiave controllano l'accesso alla chiave gestita dal cliente. Ogni chiave gestita dal cliente deve avere esattamente una policy della chiave, che contiene istruzioni che determinano chi può usare la chiave e come la possono usare. Quando crei la chiave gestita dal cliente, puoi specificare una policy della chiave. Per ulteriori informazioni, consulta [Gestione dell'accesso alle chiavi gestite dal cliente](#) nella Guida per gli sviluppatori di AWS Key Management Service .

Per utilizzare la tua chiave gestita dai clienti con le tue risorse Amazon Location, le seguenti operazioni API devono essere consentite nella politica chiave:

- [kms:CreateGrant](#): aggiunge una concessione a una chiave gestita dal cliente. Concede l'accesso di controllo a una chiave KMS specificata, che consente l'accesso alle operazioni di [concessione richieste da Amazon Location](#). Per ulteriori informazioni sull'[utilizzo di Grants](#), consulta la AWS Key Management Service Guida per gli sviluppatori.

Ciò consente ad Amazon Location di effettuare le seguenti operazioni:

- Chiama `GenerateDataKeyWithoutPlainText` per generare una chiave dati crittografata e archivarla, poiché la chiave dati non viene utilizzata immediatamente per crittografare.
- Chiama `Decrypt` per utilizzare la chiave dati crittografata memorizzata per accedere ai dati crittografati.
- Configura un preside in pensione per consentire al servizio di farlo `RetireGrant`.
- [kms:DescribeKey](#)— Fornisce i dettagli chiave gestiti dal cliente per consentire ad Amazon Location di convalidare la chiave.

Di seguito sono riportati alcuni esempi di policy che puoi aggiungere per Amazon Location:

```
"Statement" : [  
  {  
    "Sid" : "Allow access to principals authorized to use Amazon Location",  
    "Effect" : "Allow",  
    "Principal" : {  
      "AWS" : "*"   
    },  
  },  
]
```

```

    "Action" : [
      "kms:DescribeKey",
      "kms:CreateGrant"
    ],
    "Resource" : "*",
    "Condition" : {
      "StringEquals" : {
        "kms:ViaService" : "geo.region.amazonaws.com",
        "kms:CallerAccount" : "111122223333"
      }
    },
  },
  {
    "Sid": "Allow access for key administrators",
    "Effect": "Allow",
    "Principal": {
      "AWS": "arn:aws:iam::111122223333:root"
    },
    "Action" : [
      "kms:*"
    ],
    "Resource": "arn:aws:kms:region:111122223333:key/key_ID"
  },
  {
    "Sid" : "Allow read-only access to key metadata to the account",
    "Effect" : "Allow",
    "Principal" : {
      "AWS" : "arn:aws:iam::111122223333:root"
    },
    "Action" : [
      "kms:Describe*",
      "kms:Get*",
      "kms:List*",
      "kms:RevokeGrant"
    ],
    "Resource" : "*"
  }
]

```

Per ulteriori informazioni su come [specificare le autorizzazioni in una policy](#), consulta la Guida per gli sviluppatori di AWS Key Management Service .

Per informazioni sulla [Risoluzione dei problemi delle chiavi di accesso](#) consulta la Guida per gli sviluppatori di AWS Key Management Service .

## Specificare una chiave gestita dal cliente per Amazon Location

È possibile specificare una chiave gestita dal cliente come crittografia di secondo livello per le seguenti risorse:

- [Risorsa Tracker](#)
- [Collezione Geofence](#)

Quando crei una risorsa, puoi specificare la chiave dati inserendo un ID KMS, che Amazon Location utilizza per crittografare i dati personali identificabili memorizzati dalla risorsa.

- ID KMS: un [identificatore chiave per una chiave gestita](#) dal cliente. AWS KMS Inserisci l'ID della chiave, l'ARN della chiave, il nome dell'alias o l'ARN dell'alias.

## Contesto di crittografia di Amazon Location Service

Un [contesto di crittografia](#) è un set facoltativo di coppie chiave-valore che contengono ulteriori informazioni contestuali sui dati.

AWS KMS utilizza il contesto di crittografia come [dati autenticati aggiuntivi](#) per supportare la crittografia [autenticata](#). Quando includi un contesto di crittografia in una richiesta di crittografia dei dati, AWS KMS associa il contesto di crittografia ai dati crittografati. Per decrittografare i dati, nella richiesta deve essere incluso lo stesso contesto di crittografia.

## Contesto di crittografia di Amazon Location Service

Amazon Location utilizza lo stesso contesto di crittografia in tutte le operazioni AWS KMS crittografiche, in cui la chiave è `aws:geo:arn` e il valore è la [risorsa Amazon Resource Name \(ARN\)](#).

### Example

```
"encryptionContext": {
  "aws:geo:arn": "arn:aws:geo:us-west-2:111122223333:geofence-collection/SAMPLE-GeofenceCollection"
}
```

## Utilizzo del contesto di crittografia per il monitoraggio

Quando utilizzi una chiave simmetrica gestita dal cliente per crittografare la tua raccolta di tracker o geofence, puoi anche utilizzare il contesto di crittografia nei record e nei log di controllo per identificare come viene utilizzata la chiave gestita dal cliente. Il contesto di crittografia appare anche nei [log generati da AWS CloudTrail o Amazon CloudWatch Logs](#).

Utilizzo del contesto di crittografia per controllare l'accesso alla chiave gestita dal cliente

È possibile utilizzare il contesto di crittografia nelle policy delle chiavi e nelle policy IAM come `conditions` per controllare l'accesso alla chiave simmetrica gestita dal cliente. È possibile utilizzare i vincoli del contesto di crittografia in una concessione.

Amazon Location utilizza un vincolo di contesto di crittografia nelle concessioni per controllare l'accesso alla chiave gestita dal cliente nel tuo account o nella tua regione. Il vincolo della concessione richiede che le operazioni consentite dalla concessione utilizzino il contesto di crittografia specificato.

## Example

Di seguito sono riportati alcuni esempi di istruzioni delle policy delle chiavi per concedere l'accesso a una chiave gestita dal cliente per un contesto di crittografia specifico. Questa istruzione della policy impone come condizione che le concessioni abbiano un vincolo che specifica il contesto di crittografia.

```
{
  "Sid": "Enable DescribeKey",
  "Effect": "Allow",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:role/ExampleReadOnlyRole"
  },
  "Action": "kms:DescribeKey",
  "Resource": "*"
},
{
  "Sid": "Enable CreateGrant",
  "Effect": "Allow",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:role/ExampleReadOnlyRole"
  },
  "Action": "kms:CreateGrant",
  "Resource": "*",
  "Condition": {
    "StringEquals": {
```



```

      "kms:EncryptionContext:aws:geo:arn": "arn:aws:geo:us-
west-2:111122223333:tracker/SAMPLE-Tracker"
    }
  }
}

```

## Monitoraggio delle chiavi di crittografia per Amazon Location Service

Quando utilizzi una chiave gestita AWS KMS dal cliente con le tue risorse Amazon Location Service, puoi utilizzare [AWS CloudTrail](#) [Amazon CloudWatch Logs](#) per tracciare le richieste a cui Amazon Location invia. AWS KMS

Gli esempi seguenti sono AWS CloudTrail eventi per CreateGrant

GenerateDataKeyWithoutPlainTextDecrypt, e per DescribeKey monitorare le operazioni KMS chiamate da Amazon Location per accedere ai dati crittografati dalla chiave gestita dal cliente:

### CreateGrant

Quando utilizzi una chiave gestita AWS KMS dal cliente per crittografare le tue risorse di raccolta di tracker o geofence, Amazon Location invia una CreateGrant richiesta per tuo conto per accedere alla chiave KMS nel tuo account. AWS La concessione creata da Amazon Location è specifica per la risorsa associata alla chiave gestita AWS KMS dal cliente. Inoltre, Amazon Location utilizza l'RetireGrantoperazione per rimuovere una concessione quando elimini una risorsa.

L'evento di esempio seguente registra l'operazione CreateGrant:

```

{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "AROAIQDTESTANDEXAMPLE:Sampleuser01",
    "arn": "arn:aws:sts::111122223333:assumed-role/Admin/Sampleuser01",
    "accountId": "111122223333",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE3",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "AROAIQDTESTANDEXAMPLE:Sampleuser01",
        "arn": "arn:aws:sts::111122223333:assumed-role/Admin/Sampleuser01",
        "accountId": "111122223333",
        "userName": "Admin"
      }
    }
  }
}

```

```

    },
    "webIdFederationData": {},
    "attributes": {
      "mfaAuthenticated": "false",
      "creationDate": "2021-04-22T17:02:00Z"
    }
  },
  "invokedBy": "geo.amazonaws.com"
},
"eventTime": "2021-04-22T17:07:02Z",
"eventSource": "kms.amazonaws.com",
"eventName": "CreateGrant",
"awsRegion": "us-west-2",
"sourceIPAddress": "172.12.34.56",
"userAgent": "ExampleDesktop/1.0 (V1; OS)",
"requestParameters": {
  "retiringPrincipal": "geo.region.amazonaws.com",
  "operations": [
    "GenerateDataKeyWithoutPlaintext",
    "Decrypt",
    "DescribeKey"
  ],
  "keyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE",
  "granteePrincipal": "geo.region.amazonaws.com"
},
"responseElements": {
  "grantId":
"0ab0ac0d0b000f00ea00cc0a0e00fc00bce000c000f0000000c0bc0a0000aaafSAMPLE"
},
"requestID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
"eventID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
"readOnly": false,
"resources": [
  {
    "accountId": "111122223333",
    "type": "AWS::KMS::Key",
    "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
  }
],
"eventType": "AwsApiCall",
"managementEvent": true,
"eventCategory": "Management",

```

```

    "recipientAccountId": "111122223333"
  }

```

## GenerateDataKeyWithoutPlainText

Quando abiliti una chiave gestita AWS KMS dal cliente per la tua risorsa di raccolta di tracker o geofence, Amazon Location crea una chiave di tabella univoca. Invia una `GenerateDataKeyWithoutPlainText` richiesta a AWS KMS cui specifica la chiave gestita dal AWS KMS cliente per la risorsa.

L'evento di esempio seguente registra l'operazione `GenerateDataKeyWithoutPlainText`:

```

{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AWSService",
    "invokedBy": "geo.amazonaws.com"
  },
  "eventTime": "2021-04-22T17:07:02Z",
  "eventSource": "kms.amazonaws.com",
  "eventName": "GenerateDataKeyWithoutPlaintext",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "172.12.34.56",
  "userAgent": "ExampleDesktop/1.0 (V1; OS)",
  "requestParameters": {
    "encryptionContext": {
      "aws:geo:arn": "arn:aws:geo:us-west-2:111122223333:geofence-collection/SAMPLE-GeofenceCollection"
    },
    "keySpec": "AES_256",
    "keyId": "arn:aws:kms:us-west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
  },
  "responseElements": null,
  "requestID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
  "eventID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
  "readOnly": true,
  "resources": [
    {
      "accountId": "111122223333",
      "type": "AWS::KMS::Key",
      "ARN": "arn:aws:kms:us-west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
    }
  ]
}

```

```

    }
  ],
  "eventType": "AwsApiCall",
  "managementEvent": true,
  "eventCategory": "Management",
  "recipientAccountId": "111122223333",
  "sharedEventID": "57f5dbee-16da-413e-979f-2c4c6663475e"
}

```

## Decrypt

Quando accedi a un tracker crittografato o a una raccolta di geofence, Amazon Location chiama l'operazione Decrypt per utilizzare la chiave dati crittografata memorizzata per accedere ai dati crittografati.

L'evento di esempio seguente registra l'operazione Decrypt:

```

{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AWSService",
    "invokedBy": "geo.amazonaws.com"
  },
  "eventTime": "2021-04-22T17:10:51Z",
  "eventSource": "kms.amazonaws.com",
  "eventName": "Decrypt",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "172.12.34.56",
  "userAgent": "ExampleDesktop/1.0 (V1; OS)",
  "requestParameters": {
    "encryptionContext": {
      "aws:geo:arn": "arn:aws:geo:us-west-2:111122223333:geofence-collection/SAMPLE-GeofenceCollection"
    },
    "keyId": "arn:aws:kms:us-west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE",
    "encryptionAlgorithm": "SYMMETRIC_DEFAULT"
  },
  "responseElements": null,
  "requestID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
  "eventID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
  "readOnly": true,
  "resources": [

```

```

    {
      "accountId": "111122223333",
      "type": "AWS::KMS::Key",
      "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
    }
  ],
  "eventType": "AwsApiCall",
  "managementEvent": true,
  "eventCategory": "Management",
  "recipientAccountId": "111122223333",
  "sharedEventID": "dc129381-1d94-49bd-b522-f56a3482d088"
}

```

## DescribeKey

Amazon Location utilizza l'DescribeKey operazione per verificare se la chiave gestita AWS KMS dal cliente associata alla tua raccolta di tracker o geofence esiste nell'account e nella regione.

L'evento di esempio seguente registra l'operazione DescribeKey:

```

{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "AROAIQDTESTANDEXAMPLE:Sampleuser01",
    "arn": "arn:aws:sts::111122223333:assumed-role/Admin/Sampleuser01",
    "accountId": "111122223333",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE3",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "AROAIQDTESTANDEXAMPLE:Sampleuser01",
        "arn": "arn:aws:sts::111122223333:assumed-role/Admin/Sampleuser01",
        "accountId": "111122223333",
        "userName": "Admin"
      },
      "webIdFederationData": {},
      "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2021-04-22T17:02:00Z"
      }
    }
  },
}

```

```
    "invokedBy": "geo.amazonaws.com"
  },
  "eventTime": "2021-04-22T17:07:02Z",
  "eventSource": "kms.amazonaws.com",
  "eventName": "DescribeKey",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "172.12.34.56",
  "userAgent": "ExampleDesktop/1.0 (V1; OS)",
  "requestParameters": {
    "keyId": "00dd0db0-0000-0000-ac00-b0c000SAMPLE"
  },
  "responseElements": null,
  "requestID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
  "eventID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
  "readOnly": true,
  "resources": [
    {
      "accountId": "111122223333",
      "type": "AWS::KMS::Key",
      "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
    }
  ],
  "eventType": "AwsApiCall",
  "managementEvent": true,
  "eventCategory": "Management",
  "recipientAccountId": "111122223333"
}
```

## Ulteriori informazioni

Le seguenti risorse forniscono ulteriori informazioni sulla crittografia dei dati a riposo.

- Per ulteriori informazioni su [Concetti base di AWS Key Management Service](#), consulta la Guida per gli sviluppatori di AWS Key Management Service .
- Per ulteriori informazioni sulle [migliori pratiche di sicurezza per AWS Key Management Service](#), [consulta la Guida per](#) gli AWS Key Management Service sviluppatori.

## Crittografia dei dati in transito per Amazon Location Service

Amazon Location protegge i dati in transito, mentre viaggiano da e verso il servizio, crittografando automaticamente tutti i dati tra reti utilizzando il protocollo di crittografia Transport Layer Security (TLS) 1.2. Le richieste HTTPS dirette inviate alle API di Amazon Location Service vengono firmate utilizzando l'[algoritmo AWS Signature versione 4](#) per stabilire una connessione sicura.

## Identity and Access Management per Amazon Location Service

AWS Identity and Access Management (IAM) è uno strumento Servizio AWS che aiuta un amministratore a controllare in modo sicuro l'accesso alle AWS risorse. Gli amministratori IAM controllano chi può essere autenticato (effettuato l'accesso) e autorizzato (disporre delle autorizzazioni) a utilizzare le risorse di Amazon Location. IAM è uno Servizio AWS strumento che puoi utilizzare senza costi aggiuntivi.

### Argomenti

- [Destinatari](#)
- [Autenticazione con identità](#)
- [Gestione dell'accesso con policy](#)
- [Come funziona Amazon Location Service con IAM](#)
- [Come funziona Amazon Location Service con utenti non autenticati](#)
- [Esempi di policy basate sull'identità per Amazon Location Service](#)
- [Risoluzione dei problemi relativi all'identità e all'accesso ad Amazon Location Service](#)

## Destinatari

Il modo in cui utilizzi AWS Identity and Access Management (IAM) varia a seconda del lavoro svolto in Amazon Location.

Utente del servizio: se utilizzi il servizio Amazon Location per svolgere il tuo lavoro, l'amministratore ti fornisce le credenziali e le autorizzazioni necessarie. Man mano che utilizzi più funzionalità di Amazon Location per svolgere il tuo lavoro, potresti aver bisogno di autorizzazioni aggiuntive. La comprensione della gestione dell'accesso ti consente di richiedere le autorizzazioni corrette all'amministratore. Se non riesci ad accedere a una funzionalità in Amazon Location, consulta [Risoluzione dei problemi relativi all'identità e all'accesso ad Amazon Location Service](#).

Amministratore del servizio: se sei responsabile delle risorse di Amazon Location presso la tua azienda, probabilmente hai pieno accesso ad Amazon Location. È tuo compito determinare a quali funzionalità e risorse di Amazon Location devono accedere gli utenti del servizio. Devi inviare le richieste all'amministratore IAM per cambiare le autorizzazioni degli utenti del servizio. Esamina le informazioni contenute in questa pagina per comprendere i concetti di base relativi a IAM. Per ulteriori informazioni su come la tua azienda può utilizzare IAM con Amazon Location, consulta [Come funziona Amazon Location Service con IAM](#).

Amministratore IAM: se sei un amministratore IAM, potresti voler conoscere i dettagli su come scrivere policy per gestire l'accesso ad Amazon Location. Per visualizzare esempi di policy basate sull'identità di Amazon Location che puoi utilizzare in IAM, consulta. [Esempi di policy basate sull'identità per Amazon Location Service](#)

## Autenticazione con identità

L'autenticazione è il modo in cui accedi AWS utilizzando le tue credenziali di identità. Devi essere autenticato (aver effettuato l' Utente root dell'account AWS accesso AWS) come utente IAM o assumendo un ruolo IAM.

Puoi accedere AWS come identità federata utilizzando le credenziali fornite tramite una fonte di identità. AWS IAM Identity Center Gli utenti (IAM Identity Center), l'autenticazione Single Sign-On della tua azienda e le tue credenziali di Google o Facebook sono esempi di identità federate. Se accedi come identità federata, l'amministratore ha configurato in precedenza la federazione delle identità utilizzando i ruoli IAM. Quando accedi AWS utilizzando la federazione, assumi indirettamente un ruolo.

A seconda del tipo di utente, puoi accedere al AWS Management Console o al portale di AWS accesso. Per ulteriori informazioni sull'accesso a AWS, vedi [Come accedere al tuo Account AWS nella](#) Guida per l'Accedi ad AWS utente.

Se accedi a AWS livello di codice, AWS fornisce un kit di sviluppo software (SDK) e un'interfaccia a riga di comando (CLI) per firmare crittograficamente le tue richieste utilizzando le tue credenziali. Se non utilizzi AWS strumenti, devi firmare tu stesso le richieste. Per ulteriori informazioni sull'utilizzo del metodo consigliato per firmare autonomamente le richieste, consulta [Signing AWS API request](#) nella IAM User Guide.

A prescindere dal metodo di autenticazione utilizzato, potrebbe essere necessario specificare ulteriori informazioni sulla sicurezza. Ad esempio, ti AWS consiglia di utilizzare l'autenticazione a più fattori (MFA) per aumentare la sicurezza del tuo account. Per ulteriori informazioni, consulta [Autenticazione](#)



[a più fattori](#) nella Guida per l'utente di AWS IAM Identity Center e [Utilizzo dell'autenticazione a più fattori \(MFA\) in AWS](#) nella Guida per l'utente IAM.

## Account AWS utente root

Quando si crea un account Account AWS, si inizia con un'identità di accesso che ha accesso completo a tutte Servizi AWS le risorse dell'account. Questa identità è denominata utente Account AWS root ed è accessibile effettuando l'accesso con l'indirizzo e-mail e la password utilizzati per creare l'account. Si consiglia vivamente di non utilizzare l'utente root per le attività quotidiane. Conserva le credenziali dell'utente root e utilizzale per eseguire le operazioni che solo l'utente root può eseguire. Per un elenco completo delle attività che richiedono l'accesso come utente root, consulta la sezione [Attività che richiedono le credenziali dell'utente root](#) nella Guida per l'utente IAM.

## Identità federata

Come procedura consigliata, richiedi agli utenti umani, compresi gli utenti che richiedono l'accesso come amministratore, di utilizzare la federazione con un provider di identità per accedere Servizi AWS utilizzando credenziali temporanee.

Un'identità federata è un utente dell'elenco utenti aziendale, di un provider di identità Web AWS Directory Service, della directory Identity Center o di qualsiasi utente che accede utilizzando le Servizi AWS credenziali fornite tramite un'origine di identità. Quando le identità federate accedono Account AWS, assumono ruoli e i ruoli forniscono credenziali temporanee.

Per la gestione centralizzata degli accessi, consigliamo di utilizzare AWS IAM Identity Center. Puoi creare utenti e gruppi in IAM Identity Center oppure puoi connetterti e sincronizzarti con un set di utenti e gruppi nella tua fonte di identità per utilizzarli su tutte le tue applicazioni. Account AWS Per ulteriori informazioni su IAM Identity Center, consulta [Cos'è IAM Identity Center?](#) nella Guida per l'utente di AWS IAM Identity Center .

## Utenti e gruppi IAM

Un [utente IAM](#) è un'identità interna Account AWS che dispone di autorizzazioni specifiche per una singola persona o applicazione. Ove possibile, consigliamo di fare affidamento a credenziali temporanee invece di creare utenti IAM con credenziali a lungo termine come le password e le chiavi di accesso. Tuttavia, se si hanno casi d'uso specifici che richiedono credenziali a lungo termine con utenti IAM, si consiglia di ruotare le chiavi di accesso. Per ulteriori informazioni, consulta la pagina [Rotazione periodica delle chiavi di accesso per casi d'uso che richiedono credenziali a lungo termine](#) nella Guida per l'utente IAM.

Un [gruppo IAM](#) è un'identità che specifica un insieme di utenti IAM. Non è possibile eseguire l'accesso come gruppo. È possibile utilizzare gruppi per specificare le autorizzazioni per più utenti alla volta. I gruppi semplificano la gestione delle autorizzazioni per set di utenti di grandi dimensioni. Ad esempio, è possibile avere un gruppo denominato IAMAdmins e concedere a tale gruppo le autorizzazioni per amministrare le risorse IAM.

Gli utenti sono diversi dai ruoli. Un utente è associato in modo univoco a una persona o un'applicazione, mentre un ruolo è destinato a essere assunto da chiunque ne abbia bisogno. Gli utenti dispongono di credenziali a lungo termine permanenti, mentre i ruoli forniscono credenziali temporanee. Per ulteriori informazioni, consulta [Quando creare un utente IAM \(invece di un ruolo\)](#) nella Guida per l'utente IAM.

## Ruoli IAM

Un [ruolo IAM](#) è un'identità interna all'utente Account AWS che dispone di autorizzazioni specifiche. È simile a un utente IAM, ma non è associato a una persona specifica. Puoi assumere temporaneamente un ruolo IAM in AWS Management Console [cambiando ruolo](#). Puoi assumere un ruolo chiamando un'operazione AWS CLI o AWS API o utilizzando un URL personalizzato. Per ulteriori informazioni sui metodi per l'utilizzo dei ruoli, consulta [Utilizzo di ruoli IAM](#) nella Guida per l'utente IAM.

I ruoli IAM con credenziali temporanee sono utili nelle seguenti situazioni:

- **Accesso utente federato:** per assegnare le autorizzazioni a una identità federata, è possibile creare un ruolo e definire le autorizzazioni per il ruolo. Quando un'identità federata viene autenticata, l'identità viene associata al ruolo e ottiene le autorizzazioni da esso definite. Per ulteriori informazioni sulla federazione dei ruoli, consulta [Creazione di un ruolo per un provider di identità di terza parte](#) nella Guida per l'utente IAM. Se utilizzi IAM Identity Center, configura un set di autorizzazioni. IAM Identity Center mette in correlazione il set di autorizzazioni con un ruolo in IAM per controllare a cosa possono accedere le identità dopo l'autenticazione. Per informazioni sui set di autorizzazioni, consulta [Set di autorizzazioni](#) nella Guida per l'utente di AWS IAM Identity Center .
- **Autorizzazioni utente IAM temporanee:** un utente IAM o un ruolo può assumere un ruolo IAM per ottenere temporaneamente autorizzazioni diverse per un'attività specifica.
- **Accesso multi-account:** è possibile utilizzare un ruolo IAM per permettere a un utente (un principale affidabile) con un account diverso di accedere alle risorse nell'account. I ruoli sono lo strumento principale per concedere l'accesso multi-account. Tuttavia, con alcuni Servizi AWS, è possibile allegare una policy direttamente a una risorsa (anziché utilizzare un ruolo come proxy). Per

conoscere la differenza tra ruoli e politiche basate sulle risorse per l'accesso tra account diversi, consulta [Cross Account Resource Access in IAM nella IAM User Guide](#).

- Accesso tra servizi: alcuni Servizi AWS utilizzano funzionalità in altri. Servizi AWS Ad esempio, quando effettui una chiamata in un servizio, è comune che tale servizio esegua applicazioni in Amazon EC2 o archivi oggetti in Amazon S3. Un servizio può eseguire questa operazione utilizzando le autorizzazioni dell'entità chiamante, utilizzando un ruolo di servizio o utilizzando un ruolo collegato al servizio.
- Sessioni di accesso diretto (FAS): quando utilizzi un utente o un ruolo IAM per eseguire azioni AWS, sei considerato un preside. Quando si utilizzano alcuni servizi, è possibile eseguire un'operazione che attiva un'altra operazione in un servizio diverso. FAS utilizza le autorizzazioni del principale che chiama un Servizio AWS, combinate con la richiesta Servizio AWS per effettuare richieste ai servizi downstream. Le richieste FAS vengono effettuate solo quando un servizio riceve una richiesta che richiede interazioni con altri Servizi AWS o risorse per essere completata. In questo caso è necessario disporre delle autorizzazioni per eseguire entrambe le azioni. Per i dettagli delle policy relative alle richieste FAS, consulta la pagina [Forward access sessions](#).
- Ruolo di servizio: un ruolo di servizio è un [ruolo IAM](#) che un servizio assume per eseguire azioni per tuo conto. Un amministratore IAM può creare, modificare ed eliminare un ruolo di servizio dall'interno di IAM. Per ulteriori informazioni, consulta la sezione [Creazione di un ruolo per delegare le autorizzazioni a un Servizio AWS](#) nella Guida per l'utente IAM.
- Ruolo collegato al servizio: un ruolo collegato al servizio è un tipo di ruolo di servizio collegato a un Servizio AWS. Il servizio può assumere il ruolo per eseguire un'azione per tuo conto. I ruoli collegati al servizio vengono visualizzati nel tuo account Account AWS e sono di proprietà del servizio. Un amministratore IAM può visualizzare le autorizzazioni per i ruoli collegati ai servizi, ma non modificarle.
- Applicazioni in esecuzione su Amazon EC2: puoi utilizzare un ruolo IAM per gestire le credenziali temporanee per le applicazioni in esecuzione su un'istanza EC2 e che AWS CLI effettuano richieste API. AWS. Ciò è preferibile all'archiviazione delle chiavi di accesso nell'istanza EC2. Per assegnare un AWS ruolo a un'istanza EC2 e renderlo disponibile per tutte le sue applicazioni, crei un profilo di istanza collegato all'istanza. Un profilo dell'istanza contiene il ruolo e consente ai programmi in esecuzione sull'istanza EC2 di ottenere le credenziali temporanee. Per ulteriori informazioni, consulta [Utilizzo di un ruolo IAM per concedere autorizzazioni ad applicazioni in esecuzione su istanze di Amazon EC2](#) nella Guida per l'utente IAM.

Per informazioni sull'utilizzo dei ruoli IAM, consulta [Quando creare un ruolo IAM \(invece di un utente\)](#) nella Guida per l'utente IAM.

## Gestione dell'accesso con policy

Puoi controllare l'accesso AWS creando policy e collegandole a AWS identità o risorse. Una policy è un oggetto AWS che, se associato a un'identità o a una risorsa, ne definisce le autorizzazioni. AWS valuta queste politiche quando un principale (utente, utente root o sessione di ruolo) effettua una richiesta. Le autorizzazioni nelle policy determinano l'approvazione o il rifiuto della richiesta. La maggior parte delle politiche viene archiviata AWS come documenti JSON. Per ulteriori informazioni sulla struttura e sui contenuti dei documenti delle policy JSON, consulta [Panoramica delle policy JSON](#) nella Guida per l'utente IAM.

Gli amministratori possono utilizzare le policy AWS JSON per specificare chi ha accesso a cosa. In altre parole, quale principale può eseguire azioni su quali risorse e in quali condizioni.

Per impostazione predefinita, utenti e ruoli non dispongono di autorizzazioni. Per concedere agli utenti l'autorizzazione a eseguire operazioni sulle risorse di cui hanno bisogno, un amministratore IAM può creare policy IAM. L'amministratore può quindi aggiungere le policy IAM ai ruoli e gli utenti possono assumere i ruoli.

Le policy IAM definiscono le autorizzazioni relative a un'operazione, a prescindere dal metodo utilizzato per eseguirla. Ad esempio, supponiamo di disporre di una policy che consente l'operazione `iam:GetRole`. Un utente con tale policy può ottenere informazioni sul ruolo dall' AWS Management Console AWS CLI, dall' AWS API.

### Policy basate su identità

Le policy basate su identità sono documenti di policy di autorizzazione JSON che è possibile allegare a un'identità (utente, gruppo di utenti o ruolo IAM). Tali policy definiscono le azioni che utenti e ruoli possono eseguire, su quali risorse e in quali condizioni. Per informazioni su come creare una policy basata su identità, consulta [Creazione di policy IAM](#) nella Guida per l'utente IAM.

Le policy basate su identità possono essere ulteriormente classificate come policy inline o policy gestite. Le policy inline sono integrate direttamente in un singolo utente, gruppo o ruolo. Le politiche gestite sono politiche autonome che puoi allegare a più utenti, gruppi e ruoli nel tuo Account AWS. Le politiche gestite includono politiche AWS gestite e politiche gestite dai clienti. Per informazioni su come scegliere tra una policy gestita o una policy inline, consulta [Scelta fra policy gestite e policy inline](#) nella Guida per l'utente IAM.

## Policy basate su risorse

Le policy basate su risorse sono documenti di policy JSON che è possibile collegare a una risorsa. Gli esempi più comuni di policy basate su risorse sono le policy di attendibilità dei ruoli IAM e le policy dei bucket Amazon S3. Nei servizi che supportano policy basate sulle risorse, gli amministratori dei servizi possono utilizzarli per controllare l'accesso a una risorsa specifica. Quando è collegata a una risorsa, una policy definisce le azioni che un principale può eseguire su tale risorsa e a quali condizioni. È necessario [specificare un principale](#) in una policy basata sulle risorse. I principali possono includere account, utenti, ruoli, utenti federati o. Servizi AWS

Le policy basate sulle risorse sono policy inline che si trovano in tale servizio. Non puoi utilizzare le policy AWS gestite di IAM in una policy basata sulle risorse.

## Liste di controllo degli accessi (ACL)

Le liste di controllo degli accessi (ACL) controllano quali principali (membri, utenti o ruoli dell'account) hanno le autorizzazioni per accedere a una risorsa. Le ACL sono simili alle policy basate su risorse, sebbene non utilizzino il formato del documento di policy JSON.

Amazon S3 e Amazon VPC sono esempi di servizi che supportano gli ACL. AWS WAF Per maggiori informazioni sulle ACL, consulta [Panoramica delle liste di controllo degli accessi \(ACL\)](#) nella Guida per gli sviluppatori di Amazon Simple Storage Service.

## Altri tipi di policy

AWS supporta tipi di policy aggiuntivi e meno comuni. Questi tipi di policy possono impostare il numero massimo di autorizzazioni concesse dai tipi di policy più comuni.

- **Limiti delle autorizzazioni:** un limite delle autorizzazioni è una funzionalità avanzata nella quale si imposta il numero massimo di autorizzazioni che una policy basata su identità può concedere a un'entità IAM (utente o ruolo IAM). È possibile impostare un limite delle autorizzazioni per un'entità. Le autorizzazioni risultanti sono l'intersezione delle policy basate su identità dell'entità e i relativi limiti delle autorizzazioni. Le policy basate su risorse che specificano l'utente o il ruolo nel campo `Principal` sono condizionate dal limite delle autorizzazioni. Un rifiuto esplicito in una qualsiasi di queste policy sostituisce l'autorizzazione. Per ulteriori informazioni sui limiti delle autorizzazioni, consulta [Limiti delle autorizzazioni per le entità IAM](#) nella Guida per l'utente IAM.
- **Politiche di controllo dei servizi (SCP):** le SCP sono politiche JSON che specificano le autorizzazioni massime per un'organizzazione o un'unità organizzativa (OU) in. AWS Organizations  
AWS Organizations è un servizio per il raggruppamento e la gestione centralizzata di più Account

AWS di proprietà dell'azienda. Se abiliti tutte le funzionalità in un'organizzazione, puoi applicare le policy di controllo dei servizi (SCP) a uno o tutti i tuoi account. L'SCP limita le autorizzazioni per le entità negli account dei membri, inclusa ciascuna. Utente root dell'account AWS Per ulteriori informazioni su organizzazioni e policy SCP, consulta la pagina sulle [Policy di controllo dei servizi](#) nella Guida per l'utente di AWS Organizations .

- Policy di sessione: le policy di sessione sono policy avanzate che vengono trasmesse come parametro quando si crea in modo programmatico una sessione temporanea per un ruolo o un utente federato. Le autorizzazioni della sessione risultante sono l'intersezione delle policy basate su identità del ruolo o dell'utente e le policy di sessione. Le autorizzazioni possono anche provenire da una policy basata su risorse. Un rifiuto esplicito in una qualsiasi di queste policy sostituisce l'autorizzazione. Per ulteriori informazioni, consulta [Policy di sessione](#) nella Guida per l'utente IAM.

## Più tipi di policy

Quando più tipi di policy si applicano a una richiesta, le autorizzazioni risultanti sono più complicate da comprendere. Per scoprire come si AWS determina se consentire una richiesta quando sono coinvolti più tipi di policy, consulta [Logica di valutazione delle policy](#) nella IAM User Guide.

## Come funziona Amazon Location Service con IAM

Prima di utilizzare IAM per gestire l'accesso ad Amazon Location, scopri quali funzionalità IAM sono disponibili per l'uso con Amazon Location.

### Funzionalità IAM che puoi utilizzare con Amazon Location Service

Funzionalità IAM	Supporto Amazon Location
<a href="#">Policy basate su identità</a>	Sì
<a href="#">Policy basate su risorse</a>	No
<a href="#">Azioni di policy</a>	Sì
<a href="#">Risorse relative alle policy</a>	Sì
<a href="#">Chiavi di condizione della policy (specifica del servizio)</a>	Sì

Funzionalità IAM	Supporto Amazon Location
<a href="#">Liste di controllo degli accessi (ACL)</a>	No
<a href="#">ABAC (tag nelle policy)</a>	Sì
<a href="#">Credenziali temporanee</a>	Sì
<a href="#">Autorizzazioni del principale</a>	No
☉ <a href="#">Ruoli di servizio</a>	No
<a href="#">Ruoli collegati al servizio</a>	No

Per avere una visione di alto livello di come Amazon Location e altri AWS servizi funzionano con la maggior parte delle funzionalità IAM, consulta [AWS i servizi che funzionano con IAM nella IAM User Guide](#).

## Politiche basate sull'identità per Amazon Location

Supporta le policy basate su identità	Sì
---------------------------------------	----

Le policy basate su identità sono documenti di policy di autorizzazione JSON che è possibile allegare a un'identità (utente, gruppo di utenti o ruolo IAM). Tali policy definiscono le azioni che utenti e ruoli possono eseguire, su quali risorse e in quali condizioni. Per informazioni su come creare una policy basata su identità, consulta [Creazione di policy IAM](#) nella Guida per l'utente IAM.

Con le policy basate su identità di IAM, è possibile specificare quali operazioni e risorse sono consentite o respinte, nonché le condizioni in base alle quali le operazioni sono consentite o respinte. Non è possibile specificare l'entità principale in una policy basata sull'identità perché si applica all'utente o al ruolo a cui è associato. Per informazioni su tutti gli elementi utilizzabili in una policy JSON, consulta [Guida di riferimento agli elementi delle policy JSON IAM](#) nella Guida per l'utente di IAM.

## Esempi di policy basate sull'identità per Amazon Location

Per visualizzare esempi di politiche basate sull'identità di Amazon Location, consulta [Esempi di policy basate sull'identità per Amazon Location Service](#)

## Politiche basate sulle risorse all'interno di Amazon Location

Supporta le policy basate su risorse

No

Le policy basate su risorse sono documenti di policy JSON che è possibile collegare a una risorsa. Gli esempi più comuni di policy basate su risorse sono le policy di attendibilità dei ruoli IAM e le policy dei bucket Amazon S3. Nei servizi che supportano policy basate sulle risorse, gli amministratori dei servizi possono utilizzarli per controllare l'accesso a una risorsa specifica. Quando è collegata a una risorsa, una policy definisce le azioni che un principale può eseguire su tale risorsa e a quali condizioni. È necessario [specificare un principale](#) in una policy basata sulle risorse. I principali possono includere account, utenti, ruoli, utenti federati o. Servizi AWS

Per consentire l'accesso multi-account, puoi specificare un intero account o entità IAM in un altro account come principale in una policy basata sulle risorse. L'aggiunta di un principale multi-account a una policy basata sulle risorse rappresenta solo una parte della relazione di trust. Quando il principale e la risorsa sono diversi Account AWS, un amministratore IAM dell'account affidabile deve inoltre concedere all'entità principale (utente o ruolo) l'autorizzazione ad accedere alla risorsa. L'autorizzazione viene concessa collegando all'entità una policy basata sull'identità. Tuttavia, se una policy basata su risorse concede l'accesso a un principale nello stesso account, non sono richieste ulteriori policy basate su identità. Per ulteriori informazioni, consulta [Cross Account Resource Access in IAM](#) nella IAM User Guide.

### Azioni politiche per Amazon Location

Supporta le operazioni di policy

Sì

Gli amministratori possono utilizzare le policy AWS JSON per specificare chi ha accesso a cosa. Cioè, quale principale può eseguire azioni su quali risorse, e in quali condizioni.

L'elemento `Actions` di una policy JSON descrive le azioni che è possibile utilizzare per consentire o negare l'accesso a un criterio. Le azioni politiche in genere hanno lo stesso nome dell'operazione AWS API associata. Ci sono alcune eccezioni, ad esempio le azioni di sola autorizzazione che non hanno un'operazione API corrispondente. Esistono anche alcune operazioni che richiedono più operazioni in una policy. Queste operazioni aggiuntive sono denominate operazioni dipendenti.

Includi le operazioni in una policy per concedere le autorizzazioni a eseguire l'operazione associata.



Per visualizzare un elenco delle azioni di Amazon Location, consulta [Actions Defined by Amazon Location Service nel Service Authorization Reference](#).

Le azioni politiche in Amazon Location utilizzano il seguente prefisso prima dell'azione:

```
geo
```

Per specificare più operazioni in una sola istruzione, occorre separarle con la virgola.

```
"Action": [  
  "geo:action1",  
  "geo:action2"  
]
```

È possibile specificare più azioni tramite caratteri jolly (\*). Ad esempio, per specificare tutte le azioni che iniziano con la parola Get, includi la seguente azione:

```
"Action": "geo:Get*"
```

Per visualizzare esempi di politiche basate sull'identità di Amazon Location, consulta [Esempi di policy basate sull'identità per Amazon Location Service](#)

## Risorse relative alle policy per Amazon Location

Supporta le risorse di policy	Sì
-------------------------------	----

Gli amministratori possono utilizzare le policy AWS JSON per specificare chi ha accesso a cosa. Cioè, quale principale può eseguire operazioni su quali risorse, e in quali condizioni.

L'elemento JSON `Resource` della policy specifica l'oggetto o gli oggetti ai quali si applica l'operazione. Le istruzioni devono includere un elemento `Resource` o un elemento `NotResource`. Come best practice, specifica una risorsa utilizzando il suo [nome della risorsa Amazon \(ARN\)](#). Puoi eseguire questa operazione per azioni che supportano un tipo di risorsa specifico, note come autorizzazioni a livello di risorsa.

Per le azioni che non supportano le autorizzazioni a livello di risorsa, ad esempio le operazioni di elenco, utilizza un carattere jolly (\*) per indicare che l'istruzione si applica a tutte le risorse.

```
"Resource": "*"
```

Per visualizzare un elenco dei tipi di risorse di Amazon Location e dei relativi ARN, consulta [Resources Defined by Amazon Location Service nel Service Authorization Reference](#). Per sapere con quali azioni puoi specificare l'ARN di ogni risorsa, consulta [Actions Defined by Amazon Location Service](#).

Per visualizzare esempi di politiche basate sull'identità di Amazon Location, consulta [Esempi di policy basate sull'identità per Amazon Location Service](#)

## Chiavi delle condizioni della politica per Amazon Location

Supporta le chiavi di condizione delle policy specifiche del servizio	Sì
---	----

Gli amministratori possono utilizzare le policy AWS JSON per specificare chi ha accesso a cosa. Cioè, quale principale può eseguire azioni su quali risorse, e in quali condizioni.

L'elemento `Condition`(o blocco `Condition`) consente di specificare le condizioni in cui un'istruzione è in vigore. L'elemento `Condition` è facoltativo. Puoi compilare espressioni condizionali che utilizzano [operatori di condizione](#), ad esempio uguale a o minore di, per soddisfare la condizione nella policy con i valori nella richiesta.

Se specifichi più elementi `Condition` in un'istruzione o più chiavi in un singolo elemento `Condition`, questi vengono valutati da AWS utilizzando un'operazione AND logica. Se si specificano più valori per una singola chiave di condizione, AWS valuta la condizione utilizzando un'operazione logica. OR Tutte le condizioni devono essere soddisfatte prima che le autorizzazioni dell'istruzione vengano concesse.

Puoi anche utilizzare variabili segnaposto quando specifichi le condizioni. Ad esempio, puoi autorizzare un utente IAM ad accedere a una risorsa solo se è stata taggata con il relativo nome utente IAM. Per ulteriori informazioni, consulta [Elementi delle policy IAM: variabili e tag](#) nella Guida per l'utente di IAM.

AWS supporta chiavi di condizione globali e chiavi di condizione specifiche del servizio. Per visualizzare tutte le chiavi di condizione AWS globali, consulta le chiavi di [contesto delle condizioni AWS globali nella Guida](#) per l'utente IAM.

Per visualizzare un elenco di chiavi di condizione di Amazon Location, consulta [Condition Keys for Amazon Location Service nel Service](#) Authorization Reference. Per sapere con quali azioni e risorse puoi utilizzare una chiave di condizione, consulta [Azioni definite da Amazon Location Service](#).

Amazon Location supporta le chiavi di condizione per consentirti di consentire o negare l'accesso a geofence o dispositivi specifici nelle tue dichiarazioni politiche. Sono disponibili le seguenti chiavi di condizione:

- `geo:GeofenceIds` da utilizzare con le azioni Geofence. Il tipo è `ArrayOfString`
- `geo:DeviceIds` da utilizzare con le azioni Tracker. Il tipo è `ArrayOfString`.

Le seguenti azioni possono essere utilizzate `geo:GeofenceIds` nella tua policy IAM:

- `BatchDeleteGeofences`
- `BatchPutGeofences`
- `GetGeofence`
- `PutGeofence`

Le seguenti azioni possono essere utilizzate `geo:DeviceIds` nella tua policy IAM:

- `BatchDeleteDevicePositionHistory`
- `BatchGetDevicePosition`
- `BatchUpdateDevicePosition`
- `GetDevicePosition`
- `GetDevicePositionHistory`

#### Note

Non puoi utilizzare questi tasti di condizione con le `ListDevicePosition` azioni `BatchEvaluateGeofences` `ListGeofences`, o.

Per visualizzare esempi di politiche basate sull'identità di Amazon Location, consulta [Esempi di policy basate sull'identità per Amazon Location Service](#)

## ACL in Amazon Location

Supporta le ACL

No

Le liste di controllo degli accessi (ACL) controllano quali principali (membri, utenti o ruoli dell'account) hanno le autorizzazioni per accedere a una risorsa. Le ACL sono simili alle policy basate su risorse, sebbene non utilizzino il formato del documento di policy JSON.

## ABAC con Amazon Location

Supporta ABAC (tag nelle policy)

Sì

Il controllo dell'accesso basato su attributi (ABAC) è una strategia di autorizzazione che definisce le autorizzazioni in base agli attributi. In AWS, questi attributi sono chiamati tag. Puoi allegare tag a entità IAM (utenti o ruoli) e a molte AWS risorse. L'assegnazione di tag alle entità e alle risorse è il primo passaggio di ABAC. In seguito, vengono progettate policy ABAC per consentire operazioni quando il tag dell'entità principale corrisponde al tag sulla risorsa a cui si sta provando ad accedere.

La strategia ABAC è utile in ambienti soggetti a una rapida crescita e aiuta in situazioni in cui la gestione delle policy diventa impegnativa.

Per controllare l'accesso basato su tag, fornisci informazioni sui tag nell'[elemento condizione](#) di una policy utilizzando le chiavi di condizione `aws:ResourceTag/key-name`, `aws:RequestTag/key-name` o `aws:TagKeys`.

Se un servizio supporta tutte e tre le chiavi di condizione per ogni tipo di risorsa, il valore per il servizio è Yes (Sì). Se un servizio supporta tutte e tre le chiavi di condizione solo per alcuni tipi di risorsa, allora il valore sarà Parziale.

Per ulteriori informazioni su ABAC, consulta [Che cos'è ABAC?](#) nella Guida per l'utente IAM. Per visualizzare un tutorial con i passaggi per l'impostazione di ABAC, consulta [Utilizzo del controllo degli accessi basato su attributi \(ABAC\)](#) nella Guida per l'utente di IAM.

Per ulteriori informazioni sull'etichettatura delle risorse di Amazon Location, consulta [Taggare le risorse di Amazon Location Service](#).

Per visualizzare una policy basata sulle identità di esempio per limitare l'accesso a una risorsa basata su tag su tale risorsa, consulta [Controlla l'accesso alle risorse in base ai tag](#).

## Utilizzo di credenziali temporanee con Amazon Location

Supporta le credenziali temporanee	Sì
------------------------------------	----

Alcune Servizi AWS non funzionano quando accedi utilizzando credenziali temporanee. Per ulteriori informazioni, incluse quelle che Servizi AWS funzionano con credenziali temporanee, consulta la sezione relativa alla [Servizi AWS compatibilità con IAM nella IAM User Guide](#).

Stai utilizzando credenziali temporanee se accedi AWS Management Console utilizzando qualsiasi metodo tranne nome utente e password. Ad esempio, quando accedi AWS utilizzando il link Single Sign-On (SSO) della tua azienda, tale processo crea automaticamente credenziali temporanee. Le credenziali temporanee vengono create in automatico anche quando accedi alla console come utente e poi cambi ruolo. Per ulteriori informazioni sullo scambio dei ruoli, consulta [Cambio di un ruolo \(console\)](#) nella Guida per l'utente IAM.

È possibile creare manualmente credenziali temporanee utilizzando l'API or. AWS CLI AWS È quindi possibile utilizzare tali credenziali temporanee per accedere. AWS AWS consiglia di generare dinamicamente credenziali temporanee anziché utilizzare chiavi di accesso a lungo termine. Per ulteriori informazioni, consulta [Credenziali di sicurezza provvisorie in IAM](#).

## Autorizzazioni principali multiservizio per Amazon Location

Supports forward access sessions (FAS)	No
--	----

Quando utilizzi un utente o un ruolo IAM per eseguire azioni AWS, sei considerato un principale. Quando si utilizzano alcuni servizi, è possibile eseguire un'operazione che attiva un'altra operazione in un servizio diverso. FAS utilizza le autorizzazioni del principale che chiama un Servizio AWS, in combinazione con la richiesta Servizio AWS per effettuare richieste ai servizi downstream. Le richieste FAS vengono effettuate solo quando un servizio riceve una richiesta che richiede interazioni con altri Servizi AWS o risorse per essere completata. In questo caso è necessario disporre delle autorizzazioni per eseguire entrambe le azioni. Per i dettagli delle policy relative alle richieste FAS, consulta la pagina [Forward access sessions](#).

## Ruoli di servizio per Amazon Location

Supporta i ruoli di servizio

No

Un ruolo di servizio è un [ruolo IAM](#) che un servizio assume per eseguire operazioni per tuo conto. Un amministratore IAM può creare, modificare ed eliminare un ruolo di servizio dall'interno di IAM. Per ulteriori informazioni, consulta la sezione [Creazione di un ruolo per delegare le autorizzazioni a un Servizio AWS](#) nella Guida per l'utente IAM.

### Warning

La modifica delle autorizzazioni per un ruolo di servizio potrebbe interrompere la funzionalità di Amazon Location. Modifica i ruoli di servizio solo quando Amazon Location fornisce indicazioni in tal senso.

## Ruoli collegati ai servizi per Amazon Location

Supporta i ruoli collegati ai servizi

No

Un ruolo collegato al servizio è un tipo di ruolo di servizio collegato a un Servizio AWS. Il servizio può assumere il ruolo per eseguire un'azione per tuo conto. I ruoli collegati al servizio vengono visualizzati nel tuo account Account AWS e sono di proprietà del servizio. Un amministratore IAM può visualizzare le autorizzazioni per i ruoli collegati ai servizi, ma non modificarle.

Per ulteriori informazioni su come creare e gestire i ruoli collegati ai servizi, consulta [Servizi AWS supportati da IAM](#). Trova un servizio nella tabella che include un Yes nella colonna Service-linked role (Ruolo collegato ai servizi). Scegli il collegamento Sì per visualizzare la documentazione relativa al ruolo collegato ai servizi per tale servizio.

## Come funziona Amazon Location Service con utenti non autenticati

Molti scenari per l'utilizzo di Amazon Location Service, inclusa la visualizzazione di mappe sul Web o in un'applicazione mobile, richiedono l'accesso agli utenti che non hanno effettuato l'accesso con IAM. Per questi scenari non autenticati, hai due opzioni.

- Usa chiavi API: per concedere l'accesso a utenti non autenticati, puoi creare chiavi API che consentono l'accesso in sola lettura alle tue risorse di Amazon Location Service. Ciò è utile nel caso in cui non desideri autenticare tutti gli utenti. Ad esempio, un'applicazione web. Per ulteriori informazioni sulle chiavi API, vedere [Consentire l'accesso degli ospiti non autenticati alla tua applicazione utilizzando le chiavi API](#).
- Usa Amazon Cognito: un'alternativa alle chiavi API consiste nell'utilizzare Amazon Cognito per concedere l'accesso anonimo. Amazon Cognito ti consente di creare un'autorizzazione più completa con la policy IAM per definire cosa può essere fatto dagli utenti non autenticati. Per ulteriori informazioni sull'uso di Amazon Cognito, consulta [Consentire l'accesso di ospiti non autenticati alla tua applicazione utilizzando Amazon Cognito](#)

Per una panoramica sulla fornitura dell'accesso a utenti non autenticati, consulta [Concessione dell'accesso ad Amazon Location Service](#)

## Esempi di policy basate sull'identità per Amazon Location Service

Per impostazione predefinita, gli utenti e i ruoli non sono autorizzati a creare o modificare risorse Amazon Location. Inoltre, non possono eseguire attività utilizzando AWS Management Console, AWS Command Line Interface (AWS CLI) o AWS API. Per concedere agli utenti l'autorizzazione a eseguire operazioni sulle risorse di cui hanno bisogno, un amministratore IAM può creare policy IAM. L'amministratore può quindi aggiungere le policy IAM ai ruoli e gli utenti possono assumere i ruoli.

Per informazioni su come creare una policy basata su identità IAM utilizzando questi documenti di policy JSON di esempio, consulta [Creazione di policy IAM](#) nella Guida per l'utente di IAM.

Per dettagli sulle azioni e sui tipi di risorse definiti da Amazon Location, incluso il formato degli ARN per ciascun tipo di risorsa, consulta [Actions, Resources and Condition Keys for Amazon Location Service nel Service](#) Authorization Reference.

### Argomenti

- [Best practice per le policy](#)
- [Utilizzo della console Amazon Location](#)
- [Consentire agli utenti di visualizzare le loro autorizzazioni](#)
- [Utilizzo delle risorse di Amazon Location Service nella politica](#)
- [Autorizzazioni per l'aggiornamento delle posizioni dei dispositivi](#)
- [Politica di sola lettura per le risorse del tracker](#)

- [Politica per la creazione di geofence](#)
- [Politica di sola lettura per i geofence](#)
- [Autorizzazioni per il rendering di una risorsa cartografica](#)
- [Autorizzazioni per consentire le operazioni di ricerca](#)
- [Politica di sola lettura per i calcolatori di percorso](#)
- [Controlla l'accesso alle risorse in base alle chiavi di condizione](#)
- [Controlla l'accesso alle risorse in base ai tag](#)

## Best practice per le policy

Le politiche basate sull'identità determinano se qualcuno può creare, accedere o eliminare le risorse di Amazon Location nel tuo account. Queste azioni possono comportare costi aggiuntivi per l'Account AWS. Quando crei o modifichi policy basate su identità, segui queste linee guida e raccomandazioni:

- Inizia con le policy AWS gestite e passa alle autorizzazioni con privilegi minimi: per iniziare a concedere autorizzazioni a utenti e carichi di lavoro, utilizza le politiche gestite che concedono le autorizzazioni per molti casi d'uso comuni. AWS Sono disponibili nel tuo Account AWS Ti consigliamo di ridurre ulteriormente le autorizzazioni definendo politiche gestite dai AWS clienti specifiche per i tuoi casi d'uso. Per ulteriori informazioni, consulta [Policy gestite da AWS](#) o [Policy gestite da AWS per le funzioni dei processi](#) nella Guida per l'utente IAM.
- Applica le autorizzazioni con privilegio minimo: quando imposti le autorizzazioni con le policy IAM, concedi solo le autorizzazioni richieste per eseguire un'attività. Puoi farlo definendo le azioni che possono essere intraprese su risorse specifiche in condizioni specifiche, note anche come autorizzazioni con privilegi minimi. Per ulteriori informazioni sull'utilizzo di IAM per applicare le autorizzazioni, consulta [Policy e autorizzazioni in IAM](#) nella Guida per l'utente IAM.
- Condizioni d'uso nelle policy IAM per limitare ulteriormente l'accesso: per limitare l'accesso a operazioni e risorse puoi aggiungere una condizione alle tue policy. Ad esempio, è possibile scrivere una condizione di policy per specificare che tutte le richieste devono essere inviate utilizzando SSL. Puoi anche utilizzare le condizioni per concedere l'accesso alle azioni del servizio se vengono utilizzate tramite uno specifico Servizio AWS, ad esempio AWS CloudFormation. Per ulteriori informazioni, consulta la sezione [Elementi delle policy JSON di IAM: condizione](#) nella Guida per l'utente IAM.
- Utilizzo di IAM Access Analyzer per convalidare le policy IAM e garantire autorizzazioni sicure e funzionali: IAM Access Analyzer convalida le policy nuove ed esistenti in modo che aderiscano alla



sintassi della policy IAM (JSON) e alle best practice di IAM. IAM Access Analyzer offre oltre 100 controlli delle policy e consigli utili per creare policy sicure e funzionali. Per ulteriori informazioni, consulta [Convalida delle policy per IAM Access Analyzer](#) nella Guida per l'utente IAM.

- Richiedi l'autenticazione a più fattori (MFA): se hai uno scenario che richiede utenti IAM o un utente root nel Account AWS tuo, attiva l'MFA per una maggiore sicurezza. Per richiedere la MFA quando vengono chiamate le operazioni API, aggiungi le condizioni MFA alle policy. Per ulteriori informazioni, consulta [Configurazione dell'accesso alle API protetto con MFA](#) nella Guida per l'utente IAM.

Per maggiori informazioni sulle best practice in IAM, consulta [Best practice di sicurezza in IAM](#) nella Guida per l'utente di IAM.

## Utilizzo della console Amazon Location

Per accedere alla console Amazon Location Service, devi disporre di un set minimo di autorizzazioni. Queste autorizzazioni devono consentirti di elencare e visualizzare i dettagli sulle risorse di Amazon Location presenti nel tuo Account AWS. Se crei una policy basata sull'identità più restrittiva rispetto alle autorizzazioni minime richieste, la console non funzionerà nel modo previsto per le entità (utenti o ruoli) associate a tale policy.

Non è necessario concedere autorizzazioni minime per la console agli utenti che effettuano chiamate solo verso AWS CLI o l' AWS API. Al contrario, concedi l'accesso solo alle operazioni che corrispondono all'operazione API che stanno cercando di eseguire.

Per garantire che utenti e ruoli possano utilizzare la console Amazon Location, allega la seguente politica alle entità. Per ulteriori informazioni, consulta [Aggiunta di autorizzazioni a un utente](#) nella Guida per l'utente IAM.

La seguente politica consente di accedere alla console di Amazon Location Service, per poter creare, eliminare, elencare e visualizzare i dettagli sulle risorse di Amazon Location nel tuo AWS account.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "GeoPowerUser",
      "Effect": "Allow",
      "Action": [
        "geo:*"
      ],
    }
  ],
}
```

```

    "Resource": "*"
  }
]
}

```

In alternativa, puoi concedere autorizzazioni di sola lettura per facilitare l'accesso in sola lettura. Con le autorizzazioni di sola lettura, viene visualizzato un messaggio di errore se l'utente tenta di scrivere azioni come la creazione o l'eliminazione di risorse. Ad esempio, vedi [the section called "Politica di sola lettura per i tracker"](#)

## Consentire agli utenti di visualizzare le loro autorizzazioni

Questo esempio mostra in che modo è possibile creare una policy che consente agli utenti IAM di visualizzare le policy inline e gestite che sono collegate alla relativa identità utente. Questa politica include le autorizzazioni per completare questa azione sulla console o utilizzando l'API o a livello di codice. AWS CLI AWS

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ViewOwnUserInfo",
      "Effect": "Allow",
      "Action": [
        "iam:GetUserPolicy",
        "iam:ListGroupsWithUser",
        "iam:ListAttachedUserPolicies",
        "iam:ListUserPolicies",
        "iam:GetUser"
      ],
      "Resource": ["arn:aws:iam::*:user/${aws:username}"]
    },
    {
      "Sid": "NavigateInConsole",
      "Effect": "Allow",
      "Action": [
        "iam:GetGroupPolicy",
        "iam:GetPolicyVersion",
        "iam:GetPolicy",
        "iam:ListAttachedGroupPolicies",
        "iam:ListGroupPolicies",
        "iam:ListPolicyVersions",

```

```

        "iam:ListPolicies",
        "iam:ListUsers"
    ],
    "Resource": "*"
}
]
}

```

## Utilizzo delle risorse di Amazon Location Service nella politica

Amazon Location Service utilizza i seguenti prefissi per le risorse:

Prefisso della risorsa Amazon Location

Risorsa	Prefisso della risorsa
Risorse della mappa	map
Posiziona risorse	place-index
Risorse relative al percorso	route-calculator
Risorse di tracciamento	tracker
Risorse della collezione Geofence	geofence-collection

Utilizzate la seguente sintassi ARN:

```
arn:Partition:geo:Region:Account:ResourcePrefix/ResourceName
```

Per ulteriori informazioni sul formato degli ARN, consulta [Amazon Resource Names \(ARNs\) e AWS Service Namespaces](#).

### Examples (Esempi)

- Usa il seguente ARN per consentire l'accesso a una risorsa cartografica specificata.

```
"Resource": "arn:aws:geo:us-west-2:account-id:map/map-resource-name"
```

- Per specificare l'accesso a tutte le map risorse che appartengono a un account specifico, usa il carattere jolly (\*):

```
"Resource": "arn:aws:geo:us-west-2:account-id:map/*"
```

- Alcune azioni di Amazon Location, come quelle per la creazione di risorse, non possono essere eseguite su una risorsa specifica. In questi casi, è necessario utilizzare il carattere jolly (\*).

```
"Resource": "*"
```

Per visualizzare un elenco dei tipi di risorse di Amazon Location e dei relativi ARN, consulta [Resources Defined by Amazon Location Service nel Service Authorization Reference](#). Per sapere con quali azioni puoi specificare l'ARN di ogni risorsa, consulta [Actions Defined by Amazon Location Service](#).

## Autorizzazioni per l'aggiornamento delle posizioni dei dispositivi

Per aggiornare le posizioni dei dispositivi per più tracker, ti consigliamo di concedere a un utente l'accesso a una o più delle tue risorse di tracker. Dovrai anche consentire all'utente di aggiornare un batch di posizioni del dispositivo.

*In questo esempio, oltre a concedere l'accesso alle risorse Tracker1 e Tracker2, la seguente politica concede l'autorizzazione a utilizzare l'azione contro le risorse Tracker1 e Tracker2.*  
*geo:BatchUpdateDevicePosition*

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "UpdateDevicePositions",
      "Effect": "Allow",
      "Action": [
        "geo:BatchUpdateDevicePosition"
      ],
      "Resource": [
        "arn:aws:geo:us-west-2:account-id:tracker/Tracker1",
        "arn:aws:geo:us-west-2:account-id:tracker/Tracker2"
      ]
    }
  ]
}
```

Se desideri limitare l'utente in modo che possa aggiornare le posizioni del dispositivo solo per un dispositivo specifico, puoi aggiungere una chiave di condizione per quell'ID del dispositivo.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "UpdateDevicePositions",
      "Effect": "Allow",
      "Action": [
        "geo:BatchUpdateDevicePosition"
      ],
      "Resource": [
        "arn:aws:geo:us-west-2:account-id:tracker/Tracker1",
        "arn:aws:geo:us-west-2:account-id:tracker/Tracker2"
      ],
      "Condition": {
        "ForAllValues:StringLike": {
          "geo:DeviceIds": [
            "deviceId"
          ]
        }
      }
    }
  ]
}
```

## Politica di sola lettura per le risorse del tracker

Per creare una politica di sola lettura per tutte le risorse di tracker presenti nel tuo AWS account, devi concedere l'accesso a tutte le risorse di tracker. Ti consigliamo inoltre di concedere a un utente l'accesso alle azioni che gli consentano di ottenere la posizione del dispositivo per più dispositivi, ottenere la posizione del dispositivo da un singolo dispositivo e ottenere la cronologia delle posizioni.

In questo esempio, la seguente politica concede l'autorizzazione per le seguenti azioni:

- `geo:BatchGetDevicePosition` per recuperare la posizione di più dispositivi.
- `geo:GetDevicePosition` per recuperare la posizione di un singolo dispositivo.
- `geo:GetDevicePositionHistory` per recuperare la cronologia delle posizioni di un dispositivo.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "GetDevicePositions",
      "Effect": "Allow",
      "Action": [
        "geo:BatchGetDevicePosition",
        "geo:GetDevicePosition",
        "geo:GetDevicePositionHistory"
      ],
      "Resource": "arn:aws:geo:us-west-2:account-id:tracker/*"
    }
  ]
}
```

## Politica per la creazione di geofence

Per creare una politica che consenta a un utente di creare geofence, dovrai concedere l'accesso a azioni specifiche che consentano agli utenti di creare uno o più geofence su una raccolta di geofence.

*La politica seguente concede l'autorizzazione alle seguenti azioni sulla raccolta:*

- `geo:BatchPutGeofence` per creare più geofence.
- `geo:PutGeofence` per creare un singolo geofence.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "CreateGeofences",
      "Effect": "Allow",
      "Action": [
        "geo:BatchPutGeofence",
        "geo:PutGeofence"
      ],
      "Resource": "arn:aws:geo:us-west-2:account-id:geofence-collection/Collection"
    }
  ]
}
```

```
}
```

## Politica di sola lettura per i geofence

Per creare una politica di sola lettura per i geofence archiviati in una raccolta di geofence del tuo AWS account, dovrai concedere l'accesso alle azioni che leggono i geofence dalla raccolta di geofence che archivia i geofence.

*La politica seguente concede l'autorizzazione alle seguenti azioni sulla raccolta:*

- `geo:ListGeofences` per elencare i geofence nella raccolta di geofence specificata.
- `geo:GetGeofence` per recuperare un geofence dalla collezione di geofence.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "GetGeofences",
      "Effect": "Allow",
      "Action": [
        "geo:ListGeofences",
        "geo:GetGeofence"
      ],
      "Resource": "arn:aws:geo:us-west-2:account-id:geofence-collection/Collection"
    }
  ]
}
```

## Autorizzazioni per il rendering di una risorsa cartografica

Per concedere autorizzazioni sufficienti per il rendering delle mappe, dovrai concedere l'accesso ai riquadri della mappa, agli sprite, ai glifi e al descrittore di stile:

- `geo:GetMapTile` recupera i riquadri della mappa utilizzati per il rendering selettivo delle feature su una mappa.
- `geo:GetMapSprites` recupera il foglio sprite PNG e il documento JSON corrispondente che descrive gli offset al suo interno.
- `geo:GetMapGlyphs` recupera i glifi usati per visualizzare il testo.

- `geo:GetMapStyleDescriptor` recupera il descrittore di stile della mappa, contenente le regole di rendering.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "GetTiles",
      "Effect": "Allow",
      "Action": [
        "geo:GetMapTile",
        "geo:GetMapSprites",
        "geo:GetMapGlyphs",
        "geo:GetMapStyleDescriptor"
      ],
      "Resource": "arn:aws:geo:us-west-2:account-id:map/Map"
    }
  ]
}
```

## Autorizzazioni per consentire le operazioni di ricerca

Per creare una politica che consenta le operazioni di ricerca, devi prima concedere l'accesso alla risorsa dell'indice dei luoghi nel tuo AWS account. Dovrai inoltre concedere l'accesso alle azioni che consentono all'utente di effettuare ricerche utilizzando il testo mediante geocodifica e di eseguire ricerche utilizzando una posizione mediante geocodifica inversa.

In questo esempio, oltre a concedere l'accesso a *PlaceIndex*, la seguente politica concede anche l'autorizzazione alle seguenti azioni:

- `geo:SearchPlaceIndexForPosition` consente di cercare luoghi o punti di interesse vicini a una determinata posizione.
- `geo:SearchPlaceIndexForText` consente di cercare un indirizzo, un nome, una città o una regione utilizzando testo in formato libero.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
```



```

    "Sid": "Search",
    "Effect": "Allow",
    "Action": [
      "geo:SearchPlaceIndexForPosition",
      "geo:SearchPlaceIndexForText"
    ],
    "Resource": "arn:aws:geo:us-west-2:account-id:place-index/PlaceIndex"
  }
]
}

```

## Politica di sola lettura per i calcolatori di percorso

È possibile creare una politica di sola lettura per consentire a un utente di accedere a una risorsa di calcolo del percorso per calcolare un percorso.

In questo esempio, oltre a concedere l'accesso a *ExampleCalculator*, la seguente politica concede l'autorizzazione alla seguente operazione:

- `geo:CalculateRoute` calcola un percorso in base a una posizione di partenza, una posizione di destinazione e un elenco di posizioni dei waypoint.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "RoutesReadOnly",
      "Effect": "Allow",
      "Action": [
        "geo:CalculateRoute"
      ],
      "Resource": "arn:aws:geo:us-west-2:accountID:route-calculator/ExampleCalculator"
    }
  ]
}

```

## Controlla l'accesso alle risorse in base alle chiavi di condizione

Quando crei una policy IAM per concedere l'accesso all'uso di geofence o posizioni dei dispositivi, puoi utilizzare [gli operatori Condition](#) per un controllo più preciso sui geofence o sui dispositivi a cui

un utente può accedere. Puoi farlo includendo l'id geofence o l'id del dispositivo nell'elemento della tua policy. `Condition`

Il seguente esempio di policy mostra come creare una policy che consenta a un utente di aggiornare le posizioni dei dispositivi per un dispositivo specifico.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "UpdateDevicePositions",
      "Effect": "Allow",
      "Action": [
        "geo:BatchUpdateDevicePosition"
      ],
      "Resource": [
        "arn:aws:geo:us-west-2:account-id:tracker/Tracker"
      ],
      "Condition": {
        "ForAllValues:StringLike": {
          "geo:DeviceIds": [
            "deviceId"
          ]
        }
      }
    }
  ]
}
```

## Controlla l'accesso alle risorse in base ai tag

Quando crei una policy IAM per concedere l'accesso all'uso delle tue risorse Amazon Location, puoi utilizzare il [controllo degli accessi basato sugli attributi](#) per controllare meglio le risorse che un utente può modificare, utilizzare o eliminare. [Puoi farlo includendo le informazioni sui tag nell'elemento della tua policy per controllare l'accesso in base ai tag delle risorse.](#)

Il seguente esempio di policy mostra come creare una policy che consenta a un utente di creare geofence. *Ciò concede l'autorizzazione alle seguenti azioni per creare uno o più geofence su una raccolta di geofence denominata Collection:*

- `geo:BatchPutGeofence` per creare più geofence.

- `geo:PutGeofence` per creare un singolo geofence.

Tuttavia, questa politica utilizza l'`Conditionelemento` per concedere l'autorizzazione solo se il tag `Collection` ha il valore del nome utente di quell'utente. `Owner`

- Ad esempio, se un utente denominato `richard-roe` tenta di visualizzare una Amazon Location `Collection`, la *raccolta* deve essere etichettata `Owner=richard-roe` o `owner=richard-roe`. Altrimenti all'utente viene negato l'accesso.

#### Note

La chiave di tag di condizione `Owner` corrisponde a `Owner` e `owner` perché i nomi delle chiavi di condizione non effettuano la distinzione tra maiuscole e minuscole. Per ulteriori informazioni, consulta la sezione [Elementi delle policy JSON di IAM: condizione](#) nella Guida per l'utente di IAM.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "CreateGeofencesIfOwner",
      "Effect": "Allow",
      "Action": [
        "geo:BatchPutGeofence",
        "geo:PutGeofence"
      ],
      "Resource": "arn:aws:geo:us-west-2:account-id:geofence-collection/Collection",
      "Condition": {
        "StringEquals": {"geo:ResourceTag/Owner": "${aws:username}"}
      }
    }
  ]
}
```

Per un tutorial su [come definire le autorizzazioni per accedere alle risorse AWS in base ai tag](#), consulta la Guida per l'AWS Identity and Access Management utente.

# Risoluzione dei problemi relativi all'identità e all'accesso ad Amazon Location Service

Utilizza le seguenti informazioni per aiutarti a diagnosticare e risolvere i problemi più comuni che potresti riscontrare quando lavori con Amazon Location e IAM.

## Argomenti

- [Non sono autorizzato a eseguire un'azione in Amazon Location](#)
- [Non sono autorizzato a eseguire iam: PassRole](#)
- [Voglio consentire a persone esterne a me di accedere Account AWS alle mie risorse di Amazon Location](#)

## Non sono autorizzato a eseguire un'azione in Amazon Location

Se ricevi un errore che indica che non sei autorizzato a eseguire un'operazione, le tue policy devono essere aggiornate per poter eseguire l'operazione.

L'errore di esempio seguente si verifica quando l'utente IAM `mateojackson` prova a utilizzare la console per visualizzare i dettagli relativi a una risorsa `my-example-widget` fittizia ma non dispone di autorizzazioni `geo: GetWidget` fittizie.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
geo: GetWidget on resource: my-example-widget
```

In questo caso, la policy per l'utente `mateojackson` deve essere aggiornata per consentire l'accesso alla risorsa `my-example-widget` utilizzando l'azione `geo: GetWidget`.

Se hai bisogno di aiuto, contatta il tuo AWS amministratore. L'amministratore è la persona che ti ha fornito le credenziali di accesso.

## Non sono autorizzato a eseguire iam: PassRole

Se ricevi un messaggio di errore indicante che non sei autorizzato a eseguire l'`iam: PassRole` azione, le tue politiche devono essere aggiornate per consentirti di trasferire un ruolo ad Amazon Location.

Alcuni Servizi AWS consentono di trasferire un ruolo esistente a quel servizio invece di creare un nuovo ruolo di servizio o un ruolo collegato al servizio. Per eseguire questa operazione, è necessario disporre delle autorizzazioni per trasmettere il ruolo al servizio.

Il seguente errore di esempio si verifica quando un utente IAM denominato `marymajor` tenta di utilizzare la console per eseguire un'azione in Amazon Location. Tuttavia, l'azione richiede che il servizio disponga delle autorizzazioni concesse da un ruolo di servizio. Mary non dispone delle autorizzazioni per passare il ruolo al servizio.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
iam:PassRole
```

In questo caso, le policy di Mary devono essere aggiornate per poter eseguire l'operazione `iam:PassRole`.

Se hai bisogno di aiuto, contatta il tuo AWS amministratore. L'amministratore è la persona che ti ha fornito le credenziali di accesso.

## Voglio consentire a persone esterne a me di accedere Account AWS alle mie risorse di Amazon Location

È possibile creare un ruolo con il quale utenti in altri account o persone esterne all'organizzazione possono accedere alle tue risorse. È possibile specificare chi è attendibile per l'assunzione del ruolo. Per servizi che supportano policy basate su risorse o liste di controllo degli accessi (ACL), utilizza tali policy per concedere alle persone l'accesso alle tue risorse.

Per ulteriori informazioni, consulta gli argomenti seguenti:

- Per sapere se Amazon Location supporta queste funzionalità, consulta [Come funziona Amazon Location Service con IAM](#).
- Per sapere come fornire l'accesso alle tue risorse su tutto Account AWS ciò che possiedi, consulta [Fornire l'accesso a un utente IAM in un altro Account AWS di tua proprietà](#) nella IAM User Guide.
- Per scoprire come fornire l'accesso alle tue risorse a terze parti Account AWS, consulta [Fornire l'accesso a soggetti Account AWS di proprietà di terze parti](#) nella Guida per l'utente IAM.
- Per informazioni su come fornire l'accesso tramite la federazione delle identità, consulta [Fornire l'accesso a utenti autenticati esternamente \(Federazione delle identità\)](#) nella Guida per l'utente IAM.
- Per scoprire la differenza tra l'utilizzo di ruoli e politiche basate sulle risorse per l'accesso tra account diversi, consulta [Cross Account Resource Access in IAM nella IAM](#) User Guide.

## Risposta agli incidenti in Amazon Location Service

La sicurezza è la massima priorità in AWS. Come parte del [modello di responsabilità condivisa](#) del AWS cloud, AWS gestisce un data center e un'architettura di rete che soddisfano i requisiti delle organizzazioni più sensibili alla sicurezza. In qualità di AWS cliente, condividi la responsabilità di mantenere la sicurezza nel cloud. Ciò significa che puoi controllare la sicurezza che scegli di implementare dagli AWS strumenti e dalle funzionalità a cui hai accesso.

Stabilendo una linea di base di sicurezza che soddisfi gli obiettivi delle applicazioni eseguite nel cloud, sei in grado di rilevare deviazioni a cui puoi rispondere. Poiché la risposta agli incidenti di sicurezza può essere un argomento complesso, ti invitiamo a consultare le seguenti risorse per comprendere meglio l'impatto che la risposta agli incidenti (IR) e le tue scelte hanno sugli obiettivi aziendali: [AWS Security Incident Response Guide](#), white paper [AWS Security Best Practices](#) e [AWS Cloud Adoption Framework \(AWS CAF\)](#).

## Registrazione e monitoraggio in Amazon Location Service

La registrazione e il monitoraggio sono una parte importante della risposta agli incidenti. Consente di stabilire una linea di base di sicurezza per rilevare le deviazioni su cui indagare e a cui rispondere. Implementando la registrazione e il monitoraggio per Amazon Location Service, puoi mantenere l'affidabilità, la disponibilità e le prestazioni dei tuoi progetti e delle tue risorse.

AWS fornisce diversi strumenti che possono aiutarti a registrare e raccogliere dati per la risposta agli incidenti:

### AWS CloudTrail

Amazon Location Service si integra con AWS CloudTrail, un servizio che fornisce una registrazione delle azioni intraprese da un utente, ruolo o AWS servizio. Ciò include azioni dalla console di Amazon Location Service e chiamate programmatiche alle operazioni dell'API Amazon Location. Queste registrazioni di azioni sono chiamate eventi. Per ulteriori informazioni, consulta [Registrazione e monitoraggio di Amazon Location Service con AWS CloudTrail](#).

### Amazon CloudWatch

Puoi utilizzare Amazon CloudWatch per raccogliere e analizzare i parametri relativi al tuo account Amazon Location Service. Puoi abilitare gli CloudWatch allarmi per avvisarti se una metrica soddisfa determinate condizioni e ha raggiunto una soglia specificata. Quando crei un allarme, CloudWatch invia una notifica a un Amazon Simple Notification Service da te definito. Per ulteriori informazioni, consulta la sezione [Monitoring Amazon Location Service with Amazon CloudWatch](#).

## AWS Health Pannelli di controllo

Utilizzando [AWS Health le dashboard](#), puoi verificare lo stato del servizio Amazon Location Service. Puoi anche monitorare e visualizzare i dati storici su eventuali eventi o problemi che potrebbero influire sul tuo AWS ambiente. Per ulteriori informazioni, consulta la [Guida per l'utente AWS Health](#).

## Convalida della conformità per Amazon Location Service

Per sapere se un Servizio AWS programma rientra nell'ambito di specifici programmi di conformità, consulta Servizi AWS la sezione [Scope by Compliance Program Servizi AWS](#) e scegli il programma di conformità che ti interessa. Per informazioni generali, consulta Programmi di [AWS conformità Programmi](#) di di .

È possibile scaricare report di audit di terze parti utilizzando AWS Artifact. Per ulteriori informazioni, consulta [Scaricamento dei report in AWS Artifact](#) .

La vostra responsabilità di conformità durante l'utilizzo Servizi AWS è determinata dalla sensibilità dei dati, dagli obiettivi di conformità dell'azienda e dalle leggi e dai regolamenti applicabili. AWS fornisce le seguenti risorse per contribuire alla conformità:

- [Guide introduttive su sicurezza e conformità](#): queste guide all'implementazione illustrano considerazioni sull'architettura e forniscono passaggi per implementare ambienti di base incentrati sulla AWS sicurezza e la conformità.
- [Progettazione per la sicurezza e la conformità HIPAA su Amazon Web Services](#): questo white paper descrive in che modo le aziende possono utilizzare AWS per creare applicazioni idonee all'HIPAA.

### Note

Non Servizi AWS tutte sono idonee all'HIPAA. Per ulteriori informazioni, consulta la sezione [Riferimenti sui servizi conformi ai requisiti HIPAA](#).

- [AWS Risorse per la conformità](#): questa raccolta di cartelle di lavoro e guide potrebbe essere valida per il tuo settore e la tua località.
- [AWS Guide alla conformità dei clienti](#): comprendi il modello di responsabilità condivisa attraverso la lente della conformità. Le guide riassumono le migliori pratiche per la protezione Servizi AWS e mappano le linee guida per i controlli di sicurezza su più framework (tra cui il National Institute of

Standards and Technology (NIST), il Payment Card Industry Security Standards Council (PCI) e l'International Organization for Standardization (ISO)).

- [Valutazione delle risorse con regole](#) nella Guida per gli AWS Config sviluppatori: il AWS Config servizio valuta la conformità delle configurazioni delle risorse alle pratiche interne, alle linee guida e alle normative del settore.
- [AWS Security Hub](#)— Ciò Servizio AWS fornisce una visione completa dello stato di sicurezza interno. AWS La Centrale di sicurezza utilizza i controlli di sicurezza per valutare le risorse AWS e verificare la conformità agli standard e alle best practice del settore della sicurezza. Per un elenco dei servizi e dei controlli supportati, consulta la pagina [Documentazione di riferimento sui controlli della Centrale di sicurezza](#).
- [Amazon GuardDuty](#): Servizio AWS rileva potenziali minacce ai tuoi carichi di lavoro Account AWS, ai contenitori e ai dati monitorando l'ambiente alla ricerca di attività sospette e dannose. GuardDuty può aiutarti a soddisfare vari requisiti di conformità, come lo standard PCI DSS, soddisfacendo i requisiti di rilevamento delle intrusioni imposti da determinati framework di conformità.
- [AWS Audit Manager](#)— Ciò Servizio AWS consente di verificare continuamente l' AWS utilizzo per semplificare la gestione del rischio e la conformità alle normative e agli standard di settore.

## Resilienza in Amazon Location Service

L'infrastruttura AWS globale è costruita attorno a Regioni AWS zone di disponibilità. Regioni AWS forniscono più zone di disponibilità fisicamente separate e isolate, collegate con reti a bassa latenza, ad alto throughput e altamente ridondanti. Con le zone di disponibilità, puoi progettare e gestire applicazioni e database che eseguono automaticamente il failover tra zone di disponibilità senza interruzioni. Le zone di disponibilità sono più disponibili, tolleranti ai guasti e scalabili rispetto alle infrastrutture a data center singolo o multiplo tradizionali.

[Per ulteriori informazioni sulle zone di disponibilità, vedere Global Regioni AWS Infrastructure.AWS](#)

Oltre all'infrastruttura AWS globale, Amazon Location offre diverse funzionalità per aiutarti a supportare le tue esigenze di resilienza e backup dei dati.

## Sicurezza dell'infrastruttura in Amazon Location Service

In quanto servizio gestito, Amazon Location Service è protetto dalla sicurezza di rete AWS globale. Per informazioni sui servizi AWS di sicurezza e su come AWS protegge l'infrastruttura, consulta [AWS](#)



[Cloud Security](#). Per progettare il tuo AWS ambiente utilizzando le migliori pratiche per la sicurezza dell'infrastruttura, vedi [Infrastructure Protection](#) in Security Pillar AWS Well-Architected Framework.

Utilizzi chiamate API AWS pubblicate per accedere ad Amazon Location attraverso la rete. I client devono supportare quanto segue:

- Transport Layer Security (TLS). È richiesto TLS 1.2 ed è consigliato TLS 1.3.
- Suite di cifratura con Perfect Forward Secrecy (PFS), ad esempio Ephemeral Diffie-Hellman (DHE) o Elliptic Curve Ephemeral Diffie-Hellman (ECDHE). La maggior parte dei sistemi moderni, come Java 7 e versioni successive, supporta tali modalità.

Inoltre, le richieste devono essere firmate utilizzando un ID chiave di accesso e una chiave di accesso segreta associata a un principale IAM. O puoi utilizzare [AWS Security Token Service](#) (AWS STS) per generare credenziali di sicurezza temporanee per sottoscrivere le richieste.

## Configurazione e analisi delle vulnerabilità in Amazon Location

La configurazione e i controlli IT sono una responsabilità condivisa tra te AWS e te, nostro cliente. Per ulteriori informazioni, consulta il [modello di responsabilità AWS condivisa](#).

## Prevenzione del problema "confused deputy" tra servizi

Con "confused deputy" si intende un problema di sicurezza in cui un'entità che non dispone dell'autorizzazione per eseguire una certa operazione può costringere un'entità con più privilegi a eseguire tale operazione. Nel AWS, l'impersonificazione tra servizi può portare al confuso problema del vice. La rappresentazione tra servizi può verificarsi quando un servizio (il servizio chiamante) effettua una chiamata a un altro servizio (il servizio chiamato). Il servizio chiamante può essere manipolato per utilizzare le proprie autorizzazioni e agire sulle risorse di un altro cliente, a cui normalmente non avrebbe accesso. Per evitare ciò, AWS fornisce strumenti per poterti a proteggere i tuoi dati per tutti i servizi con entità di servizio a cui è stato concesso l'accesso alle risorse del tuo account.

Amazon Location Service non funge da servizio di chiamata per conto tuo verso altri AWS servizi, quindi in questo caso non è necessario aggiungere queste protezioni. Per ulteriori informazioni sul deputato confuso, consulta [Il problema del deputato confuso](#) nella Guida per l'AWS Identity and Access Management utente.

## Best practice di sicurezza per Amazon Location Service

Amazon Location Service offre una serie di funzionalità di sicurezza da prendere in considerazione durante lo sviluppo e l'implementazione delle proprie politiche di sicurezza. Le seguenti best practice sono linee guida generali e non rappresentano una soluzione di sicurezza completa. Poiché queste best practice potrebbero non essere appropriate o sufficienti per l'ambiente, gestiscile come considerazioni utili anziché prescrizioni.

## Best practice di sicurezza per i Detective per Amazon Location Service

Le seguenti best practice per Amazon Location Service possono aiutare a rilevare incidenti di sicurezza:

### Implementa strumenti di monitoraggio AWS

Il monitoraggio è fondamentale per la risposta agli incidenti e mantiene l'affidabilità e la sicurezza delle risorse di Amazon Location Service e delle tue soluzioni. Puoi implementare strumenti di monitoraggio tra i diversi strumenti e servizi disponibili tramite AWS per monitorare le tue risorse e gli altri servizi AWS.

Ad esempio, Amazon ti CloudWatch consente di monitorare i parametri per Amazon Location Service e di impostare allarmi per avvisarti se una metrica soddisfa determinate condizioni che hai impostato e ha raggiunto una soglia che hai definito. Quando crei un allarme, puoi impostare l'invio CloudWatch di una notifica di avviso utilizzando Amazon Simple Notification Service. Per ulteriori informazioni, consulta [the section called “Registrazione e monitoraggio”](#).

### Abilita gli strumenti di registrazione AWS

La registrazione fornisce una registrazione delle azioni intraprese da un utente, un ruolo o un servizio AWS in Amazon Location Service. Puoi implementare strumenti di registrazione, ad esempio AWS CloudTrail per raccogliere dati sulle azioni volte a rilevare attività insolite delle API.

Quando si crea un percorso, è possibile CloudTrail configurare la registrazione degli eventi. Gli eventi sono registrazioni delle operazioni sulle risorse eseguite su o all'interno di una risorsa, come la richiesta effettuata ad Amazon Location, l'indirizzo IP da cui è stata effettuata la richiesta, chi ha effettuato la richiesta, quando è stata effettuata la richiesta e dati aggiuntivi. Per ulteriori informazioni, consulta [Logging Data Events for Trails](#) nella Guida per l' AWS CloudTrail utente.

## Best practice di sicurezza preventiva per Amazon Location Service

Le seguenti best practice per Amazon Location Service possono aiutare a prevenire incidenti di sicurezza:

### Utilizza connessioni sicure

Utilizza sempre connessioni crittografate, ad esempio quelle che iniziano con, `https://` per proteggere le informazioni sensibili durante il transito.

### Implementa l'accesso alle risorse con privilegi minimi

Quando crei politiche personalizzate per le risorse di Amazon Location, concedi solo le autorizzazioni necessarie per eseguire un'attività. Si consiglia di iniziare con un set minimo di autorizzazioni e di concedere autorizzazioni aggiuntive, se necessario. L'implementazione dell'accesso con privilegi minimi è essenziale per ridurre il rischio e l'impatto che potrebbero derivare da errori o attacchi dannosi. Per ulteriori informazioni, consulta [the section called “Identity and Access Management”](#).

### Utilizza ID univoci a livello globale come ID dei dispositivi

Utilizza le seguenti convenzioni per gli ID dei dispositivi.

- Gli ID dei dispositivi devono essere univoci.
- Gli ID dei dispositivi non devono essere segreti, poiché possono essere utilizzati come chiavi esterne per altri sistemi.
- Gli ID dei dispositivi non devono contenere informazioni di identificazione personale (PII), come gli ID dei dispositivi telefonici o gli indirizzi e-mail.
- Gli ID dei dispositivi non devono essere prevedibili. Si consigliano identificatori opachi come gli UUID.

### Non includete informazioni personali nelle proprietà di posizione del dispositivo

Quando inviate aggiornamenti del dispositivo (ad esempio, utilizzando [DevicePositionUpdate](#)), non includete informazioni di identificazione personale (PII) come il numero di telefono o l'indirizzo e-mail nel `PositionProperties`

## Le migliori pratiche per Amazon Location Service

Questo argomento fornisce le best practice per aiutarti a usare Amazon Location Service.

Sebbene queste best practice possano aiutarti a sfruttare appieno Amazon Location Service, non

rappresentano una soluzione completa. È necessario seguire solo i consigli applicabili al proprio ambiente.

## Argomenti

- [Sicurezza](#)
- [Gestione delle risorse](#)
- [Gestione di costi e fatturazione](#)
- [Quote e utilizzo](#)

## Sicurezza

Per aiutare a gestire o addirittura evitare i rischi per la sicurezza, prendi in considerazione le seguenti best practice:

- Usa la federazione delle identità e i ruoli IAM per gestire, controllare o limitare l'accesso alle tue risorse di Amazon Location. Per ulteriori informazioni, consulta [Best practice IAM](#) nella Guida per l'utente di IAM.
- Segui il Principio del privilegio minimo per concedere solo l'accesso minimo richiesto alle tue risorse di Amazon Location Service. Per ulteriori informazioni, consulta [the section called “Gestione dell'accesso con policy”](#).
- Per le risorse di Amazon Location Service utilizzate nelle applicazioni Web, limita l'accesso utilizzando una condizione `aws:referer IAM`, limitando l'uso da parte di siti diversi da quelli inclusi nell'elenco consentito.
- Utilizza strumenti di monitoraggio e registrazione per tenere traccia dell'accesso e dell'utilizzo delle risorse. Per ulteriori informazioni, consulta [the section called “Registrazione e monitoraggio”](#) e [Logging Data Events for Trails nella Guida per l' AWS CloudTrail utente](#).
- Utilizza connessioni sicure, come quelle iniziali, `https://` per aumentare la sicurezza e proteggere gli utenti dagli attacchi durante la trasmissione dei dati tra il server e il browser.

Per ulteriori informazioni sulle best practice in materia di sicurezza investigativa e preventiva, consulta l'argomento on [the section called “Best practice di sicurezza”](#).

## Gestione delle risorse

Per aiutarti a gestire in modo efficace le tue risorse di localizzazione in Amazon Location Service, prendi in considerazione le seguenti best practice:

- Usa gli endpoint regionali che sono fondamentali per la tua base di utenti prevista per migliorare la loro esperienza. Per informazioni sugli endpoint regionali, consulta [Regioni di localizzazione ed endpoint di Amazon](#)
- Per le risorse che utilizzano fornitori di dati, come le risorse cartografiche e le risorse relative agli indici di posizione, assicurati di rispettare i termini di utilizzo del fornitore di dati specifico. Per ulteriori informazioni, consulta [Fornitori di dati](#).
- Riduci al minimo la creazione di risorse disponendo di una risorsa per ogni configurazione di mappa, indice dei luoghi o percorsi. All'interno di una regione, in genere è necessaria una sola risorsa per fornitore di dati o stile di mappa. La maggior parte delle applicazioni utilizza risorse esistenti e non crea risorse in fase di esecuzione.
- Quando utilizzi risorse diverse in un'unica applicazione, ad esempio una risorsa cartografica e un calcolatore di percorso, utilizza lo stesso fornitore di dati in ogni risorsa per garantire che i dati corrispondano. Ad esempio, che la geometria di un percorso creato con il calcolatore di percorso sia allineata con le strade sulla mappa disegnata utilizzando la risorsa della mappa.

## Gestione di costi e fatturazione

Per aiutarti a gestire i costi e la fatturazione, prendi in considerazione le seguenti best practice:

- Utilizza strumenti di monitoraggio, come Amazon CloudWatch, per monitorare l'utilizzo delle risorse. Puoi impostare avvisi che ti avvisano quando l'utilizzo sta per superare i limiti specificati. Per ulteriori informazioni, consulta [Creazione di un allarme di fatturazione per monitorare i costi AWS stimati](#) nella Amazon CloudWatch User Guide.

## Quote e utilizzo

Account AWS Includi quote che impostano un limite predefinito per la quantità di utilizzo. Puoi impostare allarmi per avvisarti quando l'utilizzo si avvicina al limite e richiedere un aumento di una quota, quando ne hai bisogno. Per informazioni su come utilizzare le quote, consulta i seguenti argomenti.

- [Quote di Amazon Location Service](#)
- [Utilizzato CloudWatch per monitorare l'utilizzo rispetto alle quote](#)
- [Visualizzazione delle quote di servizio e impostazione degli allarmi nella](#) Amazon User Guide. CloudWatch

Puoi creare allarmi per avisarti in anticipo quando stai per superare i tuoi limiti. Ti consigliamo di impostare allarmi per ogni quota in ciascuna delle aree in Regione AWS cui utilizzi Amazon Location. Ad esempio, puoi monitorare l'utilizzo dell'`SearchPlaceIndexForText` operazione e creare un allarme quando superi l'80% della tua quota attuale.

Quando ricevi un avviso relativo alla tua quota, devi decidere cosa fare. Potresti utilizzare risorse aggiuntive perché la tua base di clienti è cresciuta. In tal caso potresti voler richiedere un aumento della tua quota, ad esempio un aumento del 50% della quota per una chiamata API in quella regione. Oppure, forse c'è un errore nel tuo servizio che ti fa effettuare altre chiamate non necessarie verso Amazon Location. In tal caso, vorresti risolvere il problema nel tuo servizio.

## Cronologia dei documenti

La tabella seguente descrive la documentazione per Amazon Location Service. Per ricevere notifiche sugli aggiornamenti puoi abbonarti a un feed RSS.

Modifica	Descrizione	Data
<a href="#">Amazon Location Service rilascia un nuovo SDK per JavaScript</a>	Per semplificare lo sviluppo di applicazioni Amazon Location tramite front-end web, Amazon Location aggiunge un nuovo SDK open source che supporta AWS SDK per JavaScript v3, semplificando l'autenticazione e utilizzando GeoJSON. Per ulteriori informazioni, consulta <a href="#">Amazon Location SDK</a> .	6 luglio 2023
<a href="#">Amazon Location Service rilascia le chiavi API alla disponibilità generale</a>	Amazon Location aggiunge il supporto per place and route e annuncia la disponibilità generale della funzionalità delle chiavi API. Per ulteriori informazioni, consulta <a href="#">Utilizzo delle chiavi API</a> .	6 luglio 2023
<a href="#">Amazon Location Service aggiunge EventBridge eventi Amazon per gli aggiornamenti delle posizioni</a>	Amazon Location aggiunge il supporto per l'invio di eventi di aggiornamento della posizione del tracker a EventBridge. Per ulteriori informazioni, incluso come abilitare gli eventi per un tracker, consulta <a href="#">Reagire agli eventi con EventBridge</a>	6 luglio 2023

[Amazon Location aggiunge metadati ai geofence](#)

Utilizzando l'API Amazon Location, ora puoi aggiungere e proprietà dei metadati ai tuoi geofence. Questi vengono archiviati con il tuo geofence e inclusi negli eventi relativi al geofence in Amazon. EventBridge [Per ulteriori informazioni, consulta Disegnare geofence e Creare regole per gli eventi.](#)

15 giugno 2023

[Amazon Location aggiunge categorie per i luoghi](#)

Amazon Location aggiunge categorie nei risultati di ricerca dei luoghi e filtra i risultati per categoria. Per ulteriori informazioni, consulta [Categorie e filtri.](#)

15 giugno 2023

[Amazon Location introduce opinioni politiche](#)

Amazon Location aggiunge opinioni politiche a determinati stili di mappa. Per ulteriori informazioni, consulta [Opinioni politiche.](#)

23 maggio 2023

[Amazon Location presenta un nuovo sito demo ed esempi](#)

Amazon Location annuncia un nuovo sito Web che consente di accedere a demo ed esempi di Amazon Location. Per ulteriori informazioni, consulta il [sito demo di Amazon Location.](#)

3 maggio 2023



<a href="#">Amazon Location introduce percorsi più lunghi in Calculate RouteMatrix</a>	Amazon Location ora consente percorsi di lunghezza illimitata per i percorsi Route Matrix creati con il provider di dati HERE. Per ulteriori informazioni, consulta <a href="#">Pianificazione di percorsi più lunghi</a> .	24 aprile 2023
<a href="#">La documentazione di Amazon Location aggiunge differenze di funzionalità in base al fornitore di dati</a>	La documentazione di Amazon Location è stata aggiornata con informazioni sulle differenze e tra ciascun fornitore di dati nella ricerca di Maps, Places e nel routing. Per ulteriori informazioni, consulta <a href="#">Funzionalità per fornitore di dati</a> .	30 marzo 2023
<a href="#">Disponibilità generale delle mappe Open Data di Amazon Location</a>	Disponibilità generale del provider di dati e dello stile di Amazon Location Service, in OpenStreetMap base alle mappe Daylight. Per ulteriori informazioni, consulta <a href="#">Open Data</a> .	7 marzo 2023
<a href="#">Amazon Location aggiunge un nuovo metodo di autorizzazione in anteprima</a>	Amazon Location Service aggiunge le chiavi API come nuovo metodo di autorizzazione per utenti anonimi, in modalità anteprima. Per ulteriori informazioni, consulta <a href="#">Consentire l'accesso degli ospiti non autenticati alla tua applicazione utilizzando le chiavi API</a> .	23 febbraio 2023

<a href="#">Documentazione di Amazon Location aggiornata con le più recenti best practice IAM</a>	La documentazione di Amazon Location Service è stata aggiornata per soddisfare le AWS Identity and Access Management best practice più recenti. Per ulteriori informazioni, consulta <a href="#">la sezione Sicurezza in Amazon Location Service</a> .	26 gennaio 2023
<a href="#">Amazon Location Service si aggiunge GrabMaps come fornitore di dati nel sud-est asiatico</a>	Amazon Location si presenta GrabMaps come fornitore di dati nel sud-est asiatico. Per ulteriori informazioni, consulta <a href="#">GrabMaps</a> .	10 gennaio 2023
<a href="#">Mappe Open Data di Amazon Location Service in anteprima</a>	Nuovo provider di dati di localizzazione Amazon e nuovo stile aggiunti in anteprima pubblica, sulla base OpenStreetMap delle mappe Daylight. Per ulteriori informazioni, consulta <a href="#">Open Data (anteprima)</a> .	15 dicembre 2022
<a href="#">Nuovi stili di immagini satellitari HERE</a>	Sono stati aggiunti due nuovi stili di mappa per le mappe che utilizzano HERE come fornitore di dati, HERE Satellite Imagery e HERE Hybrid. Per ulteriori informazioni, consulta <a href="#">gli stili di mappa HERE</a> .	25 ottobre 2022

---

<a href="#">Unità negli indirizzi</a>	Amazon Location Service ora supporta unità negli indirizzi , ad esempio «123 Main St, Apartment 3B, Anytown, USA».	20 settembre 2022
<a href="#">Ottieni luoghi per ID</a>	Amazon Location Service ora include il supporto per trovare la posizione esatta suggerita dall'SearchPlaceIndexForSuggestions operazione che utilizza l'GetPlaceoperazione. Vedi <a href="#">Utilizzo del completamento automatico</a> .	20 settembre 2022
<a href="#">Chiavi di condizione aggiuntive per la policy IAM</a>	Amazon Location Service ora supporta chiavi di condizione e aggiuntive che consentono di impostare l'accesso per geofence o dispositivi specifici nella policy IAM. Vedi <a href="#">Condition keys</a> .	23 agosto 2022
<a href="#">Geofence circolari</a>	Amazon Location Service ora supporta i geofence definiti come un cerchio con un punto centrale e un raggio, per rilevare eventi quando i dispositivi si trovano entro una certa distanza da una posizione. Vedi <a href="#">Aggiungere geofence circolari</a> .	11 agosto 2022

---

<a href="#">Riferimento alle API combinate</a>	Amazon Location Service ora ha un'unica guida di riferimento API, anziché guide separate per ogni sottoservizio. Per ulteriori informazioni sulle API, consulta <a href="#">Amazon Location APIs</a> .	7 luglio 2022
<a href="#">Integrazione di Service Quotas</a>	Amazon Location è ora <a href="#">integrato con Service Quotas</a> e ti consente di visualizzare e gestire le tue quote tramite AWS Management Console o utilizzando il. AWS CLI	6 luglio 2022
<a href="#">Capitolo aggiornato sulla documentazione dei concetti</a>	Il <a href="#">capitolo sui concetti di Amazon Location</a> è stato aggiornato con ulteriori informazioni per gli utenti di Amazon Location.	22 aprile 2022
<a href="#">Nuovo tutorial di avvio rapido per Android</a>	È stato aggiunto un nuovo <a href="#">tutorial di avvio rapido</a> per lo sviluppo di Android con Kotlin per consentire agli sviluppatori di essere operativi rapidamente.	15 aprile 2022
<a href="#">Nuovi stili di mappa HERE</a>	Sono stati aggiunti due nuovi stili di mappa per le mappe che utilizzano HERE come fornitore di dati. Per ulteriori informazioni, consulta <a href="#">gli stili di mappa HERE</a> .	15 marzo 2022

---

<a href="#">Ristruttura la documentazione con esempi di codice e tutorial aggiunti</a>	<a href="#">Questa guida per sviluppatori è stata ristrutturata per trovare più facilmente gli argomenti, inclusi nuovi capitoli di avvio rapido ed esempi di codice.</a>	25 febbraio 2022
<a href="#">Filtro di posizione basato sulla precisione per i tracker</a>	<a href="#">Ora puoi utilizzare filtri basati sulla precisione durante la creazione di una risorsa di tracciamento.</a>	7 dicembre 2021
<a href="#">Completamento automatico per gli indici dei luoghi</a>	Ora puoi utilizzare il <a href="#">completamento automatico</a> durante la ricerca negli indici dei luoghi.	6 dicembre 2021
<a href="#">Nuovo tutorial Amplify per l'utilizzo delle mappe</a>	È disponibile un nuovo tutorial che mostra come visualizzare AWS Amplify le mappe in un'applicazione web. Il tutorial è disponibile su <a href="#">Using the Amplify library with Amazon Location Service</a> .	24 novembre 2021
<a href="#">Inserisci le estensioni delle query</a>	Amazon Location Service ora supporta l'impostazione di una lingua preferita per i risultati durante la geocodifica o la geocodifica inversa e aggiunge il fuso orario e altre informazioni ai risultati. <a href="#">Per ulteriori informazioni sulla geocodifica e la geocodifica inversa, consulta Geocodifica, geocodifica inversa e ricerca.</a>	16 novembre 2021

## [Filtraggio della posizione del tracker](#)

Amazon Location Service aggiunge una nuova funzionalità di filtraggio della posizione ai tracker che può aiutarti a controllare i costi. Questa funzionalità filtra alcuni aggiornamenti di posizione sui dispositivi prima che gli aggiornamenti vengano archiviati o valutati rispetto ai geofence. [Per ulteriori informazioni sul filtraggio della posizione, consulta Trackers.](#)

5 ottobre 2021

## [Operazioni di aggiornamento](#)

Le seguenti operazioni sono state aggiunte ai riferimenti API di Amazon Location Service: [UpdateMapUpdatePlaceIndex](#), [UpdateRouteCalculator](#), [UpdateGeofenceCollection](#), e [UpdateTracker](#).

19 luglio 2021

## [Aggiornamento del tutorial: funzioni Amazon Aurora PostgreSQL definite dall'utente](#)

È stato aggiunto un nuovo tutorial su come utilizzare [le funzioni Amazon Aurora PostgreSQL definite dall'utente con Amazon Location](#) per convalidare, pulire e arricchire i dati geospaziali.

19 luglio 2021

---

<a href="#">CloudFormation Risorse AWS</a>	Amazon Location ora supporta <a href="#">la creazione dei seguenti tipi di risorse nelle CloudFormation risorse AWS:</a> AWS::Location::Map AWS::Location::PlaceIndex AWS::Location::RouteCalculator AWS::Location::Tracker AWS::Location::TrackerConsumer AWS::Location::GeofenceCollection .	7 giugno 2021
<a href="#">Tagging di risorse</a>	Ora puoi <a href="#">aggiungere tag alle tue risorse di Amazon Location</a> per aiutarti a gestire, identificare, organizzare, cercare e filtrare le tue risorse.	1 giugno 2021
<a href="#">Disponibilità generale</a>	Versione di disponibilità generale della documentazione per gli sviluppatori di Amazon Location Service: <a href="#">regione, endpoint e quote di servizio aggiornati.</a>	1 giugno 2021
<a href="#">Immagini Esri</a>	Amazon Location ora supporta l'uso dello stile di mappa Esri: <a href="#">Esri Imagery</a> . Per ulteriori informazioni, vedere <a href="#">Esri World Imagery sul sito Web di Esri</a> .	1 giugno 2021

---

<a href="#">Calcolo dei percorsi</a>	Ora puoi <a href="#">utilizzare i calcolatori di percorso di Amazon Location</a> per calcolare i percorsi e stimare i tempi di viaggio in base alla rete up-to-date stradale e alle informazioni sul traffico in tempo reale fornite dal fornitore di dati scelto.	1 giugno 2021
<a href="#">AWS KMS Crittografia a chiave gestita dal cliente per i dati inattivi</a>	Amazon Location ora supporta l'uso di una chiave simmetrica gestita dal cliente che puoi creare, possedere e gestire per <a href="#">aggiungere un secondo livello di crittografia rispetto alla crittografia di AWS proprietà esistente</a> .	1 giugno 2021
<a href="#">Versione di anteprima pubblica</a>	Versione iniziale della documentazione di anteprima pubblica.	16 dicembre 2020
<a href="#">Aggiornamento del tutorial: visualizzazione delle mappe</a>	I tutorial per la visualizzazione delle mappe MapLibre per Android e iOS sono stati aggiornati per utilizzare l'SDK MapLibre nativo.	17 marzo 2020



# Glossario AWS

Per la terminologia AWS più recente, consultare il [glossario AWS](#) nella documentazione di riferimento per Glossario AWS.

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.